

Э.Б. Лубянская Е.Н. Лукаш

**ИНФОРМАЦИОННЫЕ СИСТЕМЫ
В ЭКОНОМИКЕ:
КУРСОВОЕ ПРОЕКТИРОВАНИЕ**

Учебное пособие



Воронеж 2017

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГБОУ ВО «Воронежский государственный
технический университет»

Э.Б. Лубянская Е.Н. Лукаш

ИНФОРМАЦИОННЫЕ СИСТЕМЫ
В ЭКОНОМИКЕ:
КУРСОВОЕ ПРОЕКТИРОВАНИЕ

Утверждено учебно-методическим советом
университета в качестве учебного пособия

Воронеж 2017

УДК 681.3 (075.8)
ББК 32.97Я7
Л 828

Лубянская Э.Б. Информационные системы в экономике: курсовое проектирование: учеб. пособие / Э.Б. Лубянская, Е.Н. Лукаш. – Воронеж: ФГБОУ ВО «Воронежский государственный технический университет», 2017. – 79 с.

В учебном пособии рассматриваются основы практического применения системы управления базами данных Microsoft Access 2010. Набор предлагаемых заданий охватывает основные приемы работы с одной из самых популярных систем управления базами данных. Пособие посвящено вопросам разработки курсовой работы, таким как проектирование баз данных, а также дается описание последовательности действий, структуры и содержание пояснительной записки.

Издание соответствует требованиям Федерального государственного образовательного стандарта высшего образования по специальности 38.05.01 «Экономическая безопасность», дисциплине «Информационные системы в экономике».

Может быть использовано для аспирантов технических вузов, изучающих дисциплины организационно-экономического цикла.

Ил. 24. Табл. 8. Библиогр.: 62 назв.

Рецензенты: кафедра экономики Воронежского
экономико-правового института
(зав. кафедрой канд. экон. наук,
доц. А.Э. Ахмедов);
канд. экон. наук, доц. Д.М. Шотыло

© Лубянская Э.Б., Лукаш Е.Н.,
2017

© ФГБОУ ВО «Воронежский
государственный технический
университет», 2017

ВВЕДЕНИЕ

Специализированные СУБД создаются для управления базами данных конкретного назначения — бухгалтерскими, складскими, банковскими и т. д. Заказные СУБД требуют существенных затрат, а их подготовка к работе и отладка занимают значительный период времени (от нескольких месяцев до нескольких лет). Однако, в отличие от промышленных, заказные СУБД в максимальной степени учитывают специфику работы заказчика (того или иного предприятия), их интерфейс обычно интуитивно понятен пользователям и не требует от них специальных знаний.

В настоящее время в разных сферах деятельности все более нуждаются в средствах, позволяющих быстро и безошибочно перерабатывать большое количество информации. Применение таких средств позволяет существенно снизить затраты и повысить эффективность работы.

Пособие содержит основные вопросы проектирования баз данных, описание последовательности действий при курсовом проектировании, состав проекта (работы), ключевые разделы пояснительной записки.

Результаты работы следует обязательно сохранять на устройствах флэш-памяти или отправлять на свой почтовый ящик после каждого занятия. Потеря файла БД не является уважительной причиной при защите сделанных работ.

При работе с различными версиями программы MS Access следует руководствоваться следующими правилами. Новая версия может работать с файлами, созданными в старой версии, но для внесения изменений в базу данных ее следует конвертировать в новую версию. Старая версия обычно не может работать с файлами, созданными в новой версии.

Можно настроить программу MS Access так, чтобы она по умолчанию сохраняла файлы в формате ранних версий.

В результате изучения дисциплины студент должен знать:

- основные понятия, свойства, классификацию и этапы развития информационных технологий и систем, современные принципы работы с информационно-коммуникационными технологиями, методы и средства управления информацией и управление с помощью информации инновационной деятельностью предприятия или организации (ОПК-1);

- основные методы, способы и средства получения, хранения, переработки информации, навыки работы с компьютером как средством управления информацией, современные принципы работы с деловой информацией (ОПК-3);

- пакеты прикладных программ для решения технико-экономических задач, планирования, прогнозирования, а также разработки и управления проектом в инновационной сфере деятельности предприятий или организаций (ПВК-1);

- корпоративные информационные системы, системы электронного документооборота, базы данных, сетевые информационные технологии и глобальную сеть Интернет для решения информационных, экономических и управленческих задач в инновационной сфере деятельности предприятий или организаций (ПВК-1);

- основные информационные технологии при разработке инновационных проектов (ПВК-2);

уметь:

- использовать информационно-коммуникационные технологии в инновационной сфере деятельности предприятий или организаций (ОПК-1);

- использовать компьютер как средство управления информацией (ОПК-3);

- основные пакеты прикладных программ для решения технико-экономических задач, планирования, прогнозирования, а также разработки и управления проектом в инновационной сфере деятельности предприятий или организаций (ПВК-1);

- использовать корпоративные информационные системы, системы электронного документооборота, базы данных, сетевые информационные технологии и глобальную сеть Интернет для решения информационных, экономических и управленческих задач в инновационной сфере деятельности предприятий или организаций (ПВК-1);

- использовать основные информационные технологии при разработке инновационных проектов (ПВК-2);

владеть:

- современными принципами работы с информационно-коммуникационными технологиями, методами и средствами управления информацией в инновационной сфере деятельности предприятий или организаций (ОПК-1);

- основными способами и средствами получения, хранения, переработки информации, навыками работы с компьютером как средством управления информацией, современными принципами работы с деловой информацией (ОПК-3);

- навыками работы с пакетами прикладных программ для решения технико-экономических задач, планирования, прогнозирования, а также разработки и управления проектом в инновационной сфере деятельности предприятий или организаций (ПВК-1);

- навыками работы с корпоративными информационными системами, системами электронного документооборота, базами данных, сетевыми информационными технологиями и глобальной сетью Интернет для решения информационных, экономических и управленческих задач в инновационной сфере деятельности предприятий или организаций (ПВК-1);

- навыками работы с информационными технологиями при разработке инновационных проектов (ПВК-2).

1. БАЗЫ ДАННЫХ

1.1. Понятия и определения

Развитие вычислительной техники послужило предпосылкой к появлению новой информационной технологии интегрированного хранения и обработки данных – концепции баз данных. Ключевым понятием здесь является банк данных – это система, содержащая специально организованные данные, а также технические, программные и языковые средства, целью которых является централизованное накопление и коллективное многоцелевое использование данных [8].

В основу данной технологии положен механизм предоставления обрабатывающей программе только тех данных, которые ей требуются. Форма представления данных также устанавливается заранее и описывается на логическом уровне. Логическая структура данных, хранящихся в базе, называется моделью представления данных.

Базой данных принято называть именованную совокупность данных, которая отображает состояние объектов и отношений между ними в рассматриваемой предметной области. Отличительной чертой БД является постоянство:

данные в БД постоянно накапливаются и используются;
структура и состав задач в рамках конкретной БД постоянны и стабильны во времени;

БД постоянно хранит актуальные данные [4].

Фактически СУБД представляет собой комплекс программных и языковых средств, который служит для создания, ведения и совместного использования БД большим числом пользователей [20].

Схема описанного представления приведена на рис. 1.1.

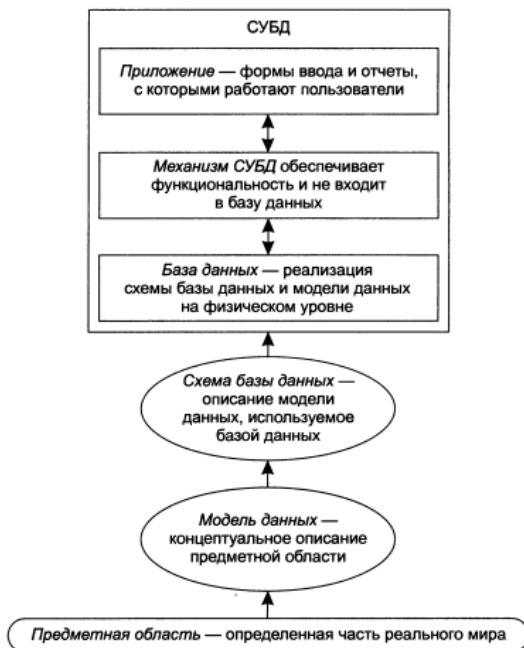


Рис. 1.1. Общая терминология БД

1.2. Компоненты системы баз данных

Основные компоненты любой системы баз данных представлены на рис. 1.2.

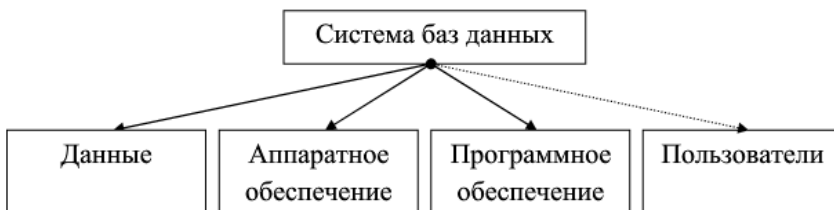


Рис. 1.2. Компоненты системы баз данных

Данные

Любая БД хранит в себе некоторый набор постоянных данных. Кроме того, существуют еще и транзитные данные, представляющие собой входные, выходные данные, а также промежуточные результаты.

Входные данные – это та информация, которая поступает в систему, например, вводится с клавиатуры. Эти данные могут являться причиной корректировки постоянных данных, но фактически не являются частью БД.

Выходные данные – это результаты и сообщения, которые выдаются системой. В большинстве случаев они берутся из постоянных данных и также не являются частью БД.

Кроме данных, характеризующих предметную область, БД хранят данные о способах и методах переработки информации или о том, где найти данные – знания о местонахождении данных, оценки этих данных, описание источников и т.п. Такие данные называются метаинформацией [6]. Централизованное хранение метаинформации представляет собой словарь данных, что отражает свойство самодокументированности БД.

Данные в любой БД должны быть общими и интегрированными. Общие данные реализуют возможность использования отдельных наборов данных из общей базы различными группами пользователей для решения каких-либо задач.

Интегрированные данные представляют собой данные, собранные из разных источников и имеющие единый способ доступа.

Эти два свойства данных являются преимуществом использования систем БД корпоративного уровня, а интеграция отражает преимущество при использовании персональных систем.

Аппаратное обеспечение

В общем виде принято выделять две группы свойств, которые наиболее важны для любой системы БД. В первую очередь сюда относятся устройства хранения данных. Затем – устройства обработки данных.

В небольших системах хранение и обработка могут быть реализованы на одном компьютере. Однако большинство крупных БД задействуют различные типы систем хранения, а также целое множество серверов, отвечающих за обработку данных.

Программное обеспечение

Под программным обеспечением (ПО) понимаются все средства, позволяющие конечным пользователям работать с данными. Кроме того, ПО решает и другие задачи, например, обеспечивает безопасность данных, реализует технологию одновременного доступа и т.п. Весь описанный комплекс программ принято называть системой управления базами данных – СУБД [19].

Основная функция СУБД – предоставлению пользователю возможности работать с БД, не вникая в детали уровня аппаратного обеспечения.

Кроме СУБД к компоненту программного обеспечения относятся и другие компоненты - различные утилиты, генераторы отчетов, а также пользовательское прикладное ПО.

Пользователи

Все пользователи БД делятся на следующие группы:

конечные пользователи – люди, работающие с БД непосредственно с терминала или рабочей станции. Эти пользователи могут использовать разработанное специально для них прикладное ПО или же встроенные средства СУБД. Именно для поддержания деятельности таких пользователей создается вся система баз данных;

прикладные программисты – люди, отвечающие за разработку и реализацию прикладных программных продуктов, взаимодействующих с БД. Разработанные продукты обращаются с запросами к СУБД, а затем получают результаты и предоставляют их пользователю. Существуют программы пакетной обработки и оперативные приложения, задачей которых является поддержка работы конечного пользователя, имеющего интерактивный доступ к системе;

администраторы данных – люди, отвечающие за данные организации. Именно администраторы данных принимают решения о том, какие данные требуется вносить в БД, а также кому и к каким данным можно иметь доступ;

администраторы баз данных – технические специалисты, отвечающие за реализацию решений администраторов данных [17].

1.3. Функции СУБД

Как говорилось ранее, основная задача СУБД – управление базой данных, которое предполагает выполнение ряда функций.

Управление словарем данных

Функционирование СУБД предусматривает хранение данных и отношений между ними в определенном словаре данных (data dictionary). Для поиска структур данных и отношений СУБД использует этот словарь, что помогает избежать кодирования в рамках каждого пользовательского приложения. Кроме того, любые корректировки данных и их взаимоотношений автоматически регистрируются в словаре данных, что также позволяет избежать необходимости модификации ПО. Работа со словарем достигается благодаря абстракции данных, что устраняет структурную зависимость, а также зависимость по данным.

Управление хранением данных

В рамках СУБД создаются специальные сложные структуры для хранения данных, что освобождает программистов от определения и программирования физических свойств обрабатываемых данных. Существующие на сегодняшний день СУБД способны хранить не только данные, но и связанные с ними экранные формы, схемы отчетов, правила проверки данных и т.п.

Представление и преобразование данных

СУБД отвечает за структурирование вводимых в БД данных, автоматически преобразуя их в форму, удобную для хранения. Это позволяет пользователю избежать процедуры преобразования данных из логического формата в физический. Кроме того, СУБД позволяет преобразовывать логические запросы в определенные команды, тем самым обеспечивая абстракцию данных и программную независимость.

Управление безопасностью

СУБД создает собственную систему безопасности, задачей которой является обеспечение конфиденциальности данных и защиту пользователя. СУБД содержит определенный набор правил безопасности, которые позволяют установить права доступа различным пользователям к различным данным.

Управление многопользовательским доступом

Чтобы обеспечить доступ сразу нескольких пользователей к одним и тем же данным в рамках СУБД создаются специальные сложные структуры. Основная задача данного подхода – обеспечение целостности и непротиворечивости данных. Для этого в СУБД используются сложные алгоритмы, которые гарантируют, что несколько пользователей одновременно смогут получить доступ к ресурсам БД без риска нарушения целостности

Управление резервным копированием и восстановлением

СУБД включает в себя набор процедур резервного копирования и восстановления данных. Такие процедуры необходимы в случаях повреждения БД, например, при физическом повреждении сектора на жестком диске или после аварийного отключения питания.

Управление целостностью данных

СУБД содержит ряд правил, которые обеспечивают целостность данных. Эти правила позволяют избежать избыточности, а также гарантируют непротиворечивость хранимых данных. Кроме того, для обеспечения целостности применя-

ются еще и связи между данными, хранящиеся в словаре данных.

Поддержка языка доступа к данным и интерфейсов прикладного программирования

СУБД обеспечивает доступ к хранящимся данным с помощью специального языка запросов. Языком запросов называется не процедурный язык, предоставляющий пользователю возможность самостоятельно определить, что именно необходимо выполнить, не указывая, как это нужно сделать. Язык запросов содержит два основных компонента:

язык определения данных (Data Definition Language, DDL) – определяет структуры размещения данных;

язык манипулирования данными (Data Manipulation Language, DML) – реализует запросы и позволяет извлекать данные из БД.

Интерфейсы взаимодействия с БД

Современные СУБД поддерживают специальное ПО, реализующее взаимодействие с БД так, чтобы она могла принимать пользовательские запросы в рамках сетевого окружения [15].

1.4. Классификация СУБД

Как говорилось ранее, логическая структура данных, хранящихся в БД, называется моделью представления этих данных. Существует несколько основных моделей представления.

Реляционная модель

Реляционная модель представляет собой простейшую и наиболее привычную форму представления данных – в табличном виде. Теория множеств имеет особый математический аппарат для работы с такой моделью – реляционное исчисление и реляционную алгебру. Главным достоинством этой модели является простота инструментальных средств поддержки, а

недостатком – жесткость структуры данных, а также зависимость скорости работы с БД от ее размеров.

В основе реляционной модели лежит понятие отношения. Отношением называется множество элементов, называемых кортежами. В привычном понимании кортежем является строка таблицы. Столбец таблицы является атрибутом. Базовая структура данной модели построена на декартовом произведении доменов. Доменом называется множество значений, которое может принимать элемент данных [9].

Существует еще и постреляционная модель данных, расширяющая описанную. Она снимает ограничение неделимости данных, которые хранятся в записях таблиц. Кроме того, постреляционная модель допускает использование многозначных полей, при этом набор значений таких полей считается отдельной самостоятельной таблицей, которая встроена в основную.

Иерархическая модель

Иерархическая модель данных представляет собой дерево (рис. 1.3).

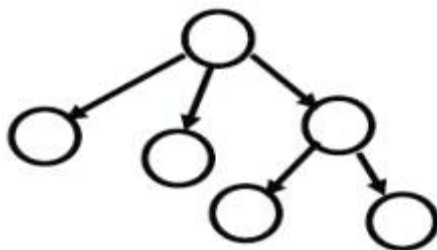


Рис. 1.3. Иерархическая модель

Каждое такое дерево состоит из набора типов, одним из которых является корень. Корневым называется такой тип, который имеет подчиненные типы, при этом сам не является подтипом. Подтип – потомок по отношению к типу-предку.

Потомки одного типа являются близнецами по отношению друг к другу.

Каждый тип, входящий в дерево, представляет собой простой или составной тип записи. Поля записи (атрибуты) хранят символьные или числовые значения, которые и являются содержимым БД.

Преимуществом иерархической модели является эффективное использование памяти ЭВМ, а также высокие показатели времени выполнения основных операций над данными.

Основной недостаток – громоздкость в плане обработки информации со сложными логическими связями, а также сложность для понимания.

Сетевая модель

Сетевая модель отображает отношения элементов БД в виде графа (рис. 1.4).

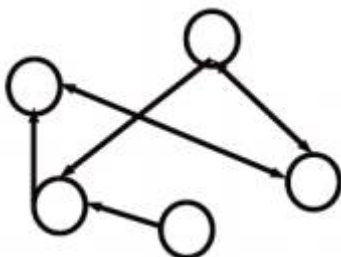


Рис. 1.4. Сетевая модель

Основными элементами данной модели являются записи и связи. На формирование связи не накладывается особых ограничений – записи-потомки могут иметь неограниченное число записей-предков.

Преимущество данной модели заключается в возможности эффективной реализации по показателям оперативности и затрат памяти.

Недостатки – высокая сложность и жесткость схемы БД. Кроме того, в сетевой модели ослаблен контроль целостности

связей из-за допустимости установления произвольных отношений [1].

1.5. Краткие выводы

В данной главе были рассмотрены основные понятия теории баз данных, описаны компоненты системы БД, выделены функции системы управления базами данных, а также приведена классификация СУБД в зависимости от используемой модели данных.

2. СУБД MS ACCESS

2.1. Общие сведения

MS Access является программой создания и редактирования БД, входящей в пакет офисных программ Microsoft Office.

MS Access объединяет сведения из различных источников в рамках одной реляционной БД. Кроме того, данная СУБД позволяет создавать формы, отчеты и запросы, что помогает эффективно обновлять данные, осуществлять поиск, получать необходимые выборки данных, печатать отчеты, диаграммы и т.п. Все компоненты БД при этом хранятся в едином файле с расширением ACCDB [3].

Основными объектами данной СУБД являются следующие:

таблица – классическая двумерная таблица, задачей которой является хранение данных в реляционной БД. Данные хранятся в записях (строках таблицы), состоящих из отдельных полей - атрибутов (столбцов). Каждая таблица хранит информацию определенного типа в зависимости от описываемой сущности;

запрос – средство отбора данных по какому-либо условию. При помощи запросов реализуется выбор необходимых данных из БД;

форма – средство для упрощения ввода и редактирования данных таблиц БД;

отчет – средство, позволяющее извлекать нужную информацию из БД, представляя ее в удобном для восприятия виде. Кроме того, данное средство автоматически генерирует документ для распечатки;

страницы – страницы доступа к данным, представляющие собой web-страницы для работы с БД посредством сети Интернет;

макрос – набор макрокоманд, которые создаются пользователем с целью автоматизации каких-либо операций с БД;

модуль – объект, содержащий программы, написанные на языке Visual Basic. Данные программы обычно применяются при обработке данных;

область со списком – список возможных режимов создания объектов [1].

2.2. Нормализация отношений

Нормализация представляет собой формальный метод анализа отношений, в основе которого лежат ключи и функциональные зависимости. В теории реляционных БД существует ряд нормальных форм.

Первая нормальная форма

Отношение находится в первой нормальной форме в том случае, если все его атрибуты имеют атомарные значения. Это говорит о том, что каждое поле содержит ровно одно значение. Пример приведения отношения к первой нормальной форме приведен в табл. 2.1, 2. 2.

Таблица 2.1

Ненормализованная таблица

Имя	Пол
{Александр, Николай}	Мужской
Елена	Женский

Таблица 2.2

Отношение в первой нормальной форме

Имя	Пол
Александр	Мужской
Николай	Мужской
Елена	Женский

Вторая нормальная форма

Отношение находится во второй нормальной форме, если оно находится в первой нормальной форме, и каждый атрибут функционально зависит от ключевого поля.

Третья нормальная форма

Отношение находится в третьей нормальной форме, если оно находится во второй нормальной форме, и ни один из атрибутов не является транзитивно зависимым от ключа.

В большинстве случаев транзитивная зависимость вызвана наличием в отношении двух семантических зависимостей различных типов.

Обычно для создания БД достаточно трех описанных форм.

2.3. Проектирование БД

Этап проектирования БД предполагает определение логической структуры данных для выбранной предметной области. Логическая структура определяет состав таблиц и их взаимосвязи.

Описание предметной области БД должно охватывать реальные процессы и объекты, а также определять все необходимые источники информации для выполнения предполагаемых запросов пользователей.

Состав и структура данных определяются при помощи анализа предметной области. Структура данных может быть представлена в виде информационно-логической модели (ИЛМ). При разработке модели можно использовать два подхода:

аналитический (процессный) – при таком подходе в первую очередь формируется список задач, решение которых должно быть реализовано в рамках БД, затем выявляются все информационные потребности, и только после этого определяются состав, структура объектов модели и связи между ними;

интуитивный – при данном подходе сразу устанавливаются типовые информационные объекты предметной области и связи между ними.

На практике оба этих подхода сочетаются для достижения лучшего эффекта. Это обусловлено тем, что на начальном этапе в большинстве случаев разработчики не располагают исчерпывающими сведениями обо всех задачах. Кроме того, гибкие средства создания реляционной БД позволяют на любом этапе разработки вносить корректировки в структуру, не нарушая ранее введенных данных.

Этапы проектирования и создания БД Access отображены на рис. 2.1.

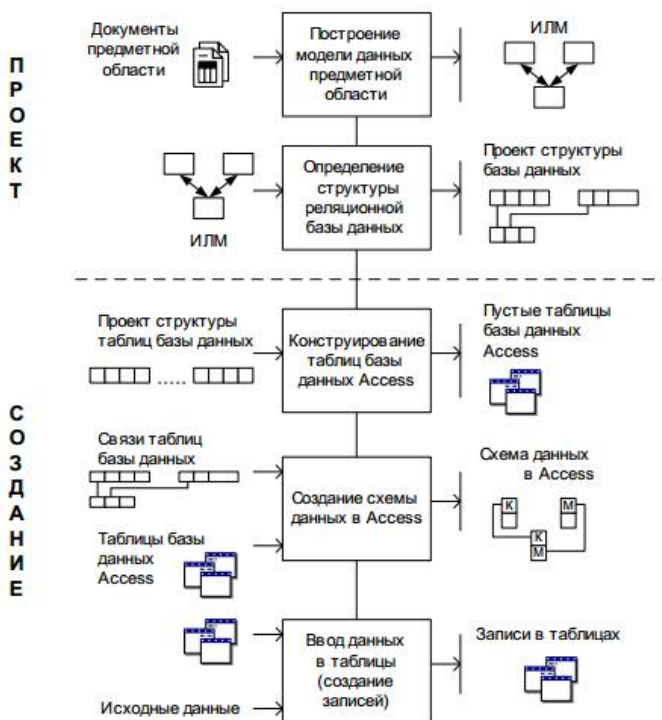


Рис. 2.1. Этапы проектирования и создания БД

Главным элементом описанной структуры является информационный объект, который соответствует всем требованиям нормализации БД. Информационный объект адекватно отображается в виде реляционной таблицы. Связи между объектами соответствуют логическим связям между парой таблиц. Эти связи устанавливаются по уникальному ключу главной таблицы. При этом поле связи подчиненной таблицы может быть либо частью уникального ключа, либо вовсе в него не входить.

Процесс создания БД начинается с конструирования таблиц. Для поддержания условия целостности каждая таблица должна содержать ключевое поле, а также правила проверки значений полей.

Существует несколько видов ключей:

первичный ключ – одно или несколько полей, однозначно идентифицирующих запись. Данный ключ применяется для связывания таблиц с внешними ключами в других таблицах. Первичный ключ, который содержит информационные поля таблицы и несет полезную информацию об объектах, называется естественным. Ключ, который реализован в виде дополнительного служебного поля – искусственный (суррогатный) – представляет собой числовое поле, где хранятся числа возрастающей последовательности;

внешний ключ – одно или несколько полей, которые содержат ссылку на поля первичного ключа другой таблицы. Внешний ключ отражает способ связи таблиц.

Связи между таблицами также бывают нескольких видов:

один к одному – каждой записи первой сущности соответствует ровно одна запись второй сущности, и наоборот;

один ко многим – каждой записи первой сущности может соответствовать несколько записей второй сущности, но каждой записи второй сущности соответствует только одна запись первой сущности;

многие ко многим - каждой записи первой сущности может соответствовать несколько записей второй сущности, и наоборот [18].

Следующим этапом является создание схемы данных, отображающей связи между таблицами. В схеме данных также могут быть заданы дополнительные параметры поддержания связной целостности данных.

Связная целостность данных отражает факт того, что в БД установлены связи, которые поддерживаются в процессах загрузки, добавления и удаления записей в связанных таблицах, а также при изменении значений ключевых полей. При таком подходе подчиненные таблицы не могут содержать таких записей, для которых нет связанной записи в главной таблице.

После создания схемы данных начинается этап загрузки – ввода информации в БД. В большинстве случаев для этого применяются экранные формы, выступающие в качестве пользовательского интерфейса.

Проектирование БД, в основе которого лежит построение нормализованной модели данных предметной области, позволяет легко получить логическую структуру реляционной БД, где автоматически поддерживаются непротиворечивость и целостность данных [2].

2.4. Физическая структура данных

Физическую структуру данных определяет тип данных каждого поля таблицы. MS Access использует типы данных, приведенные в табл. 2.3.

Таблица 2.3

Типы данных

Тип	Описание
Текстовый	Алфавитно-цифровые знаки для представления текста. Не применим в вычислениях. Не превосходит 255 символов.
Мемо	Алфавитно-цифровые знаки для представления форматированного текста длиннее 255 символов
Числовой	Числовые данные, участвующие в выражениях
Дата/время	Хранение значений даты и времени
Денежный	Хранений валютных величин
Счетчик	Уникальное числовое значение, автоматически инкрементирующееся
Логический	Поля, имеющие значения «Да» или «Нет»
OLE	Хранение OLE-объектов других приложений MS Windows
Вложение	Рисунки, изображения, бинарные файлы, файлы MS Office
Гиперссылка	Хранение гиперссылок вызова web-страниц, а также ссылок на объекты Access
Мастер подстановок	Поле, реализующее выбор значений из других таблиц, запросов или списка значений

Кроме типа данных, также можно задать ряд свойств поля, которые отражены в табл. 2.4. При помощи значений свойств полей реализуется управление отображением данных, предотвращается ввод ошибочных значений, задаются значения по умолчанию, ускоряются процессы сортировки и поиска, а также происходит управление другими функциональными характеристиками и внешним видом полей [13].

Таблица 2.4

Свойства полей

Свойство	Назначение
Размер поля	Максимальный размер данных для текстового, числового типа, а также типа «счетчик»
Формат	Формат данных для отображения и печати
Число десятичных знаков	Число отображаемых знаков в дробной части числовых значений
Новые значения	Способ присвоения значений полю «счетчик» (последовательный инкремент или случайные числа)
Маска ввода	Отображение знаков управления вводом данных
Подпись	Текст по умолчанию в формах, отчетах и запросах
Значение по умолчанию	Автоматическое значение поля при добавлении новой записи
Условие на значение	Ограничения целостности, проверяемые при вводе данных
Сообщение об ошибке	Текст, отображаемый при нарушении значения правила «Условие на значение»
Индексированное поле	Ускорение доступа к данным этого поля при помощи создания и применения индекса

2.5. Краткие выводы

В данной главе было дано описание СУБД MS Access, выделены ее основные объекты, описаны основные нормальные формы, используемые при проектировании реляционных БД. Кроме того, приводится описание этапа проектирования БД, а также физической структуры данных, реализованной в рамках рассматриваемой СУБД.

3. ЦЕЛЬ И ЗАДАЧИ ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ

Студент при выполнении курсовой работы должен:

- рассмотреть современные информационные технологии, в том числе программное и аппаратное обеспечение, и показать возможности их применения в сфере экономики;
- показать умение работать с различными источниками информации, в том числе с Интернетом;
- показать владение навыками работы в современных офисных пакетах прикладных программ.

4. ОСНОВНЫЕ ТРЕБОВАНИЯ, ПРЕДЪЯВЛЯЕМЫЕ К КУРСОВОЙ РАБОТЕ

Курсовая работа выполняется в соответствии с выбранной темой и требованиями, представленными в данном разделе.

Тема работы выбирается студентом самостоятельно из списка в 8 разделе данного учебного пособия.

В одной учебной группе недопустимо повторение тем. Допускается использование других тем, самостоятельно сформулированных студентом, но только при согласовании с преподавателем.

Курсовая работа должна содержать титульный лист, содержание, введение, основную часть, заключение, список использованной литературы, приложения.

Страницы и основные разделы должны быть пронумерованы.

Рисунки и таблицы должны иметь названия и быть пронумерованы.

Работа должна быть выполнена с использованием текстового процессора Microsoft Word версии 2010 (в виде одного файла формата docx) и сдается на проверку в распечатанном и электронном виде на CD или DVD диске. Диск должен быть

подписан - указаны номер группы, фамилия и инициалы студента, название предмета и номер темы.

Имя файла должно иметь вид:

№группы_ФамилияИО_предмет_ККР№темы.docx.

Например:

ЭБ_161_ИвановАС_ИСЭ_ККР56.docx.

Кроме того, должна быть разработана и представлена презентация курсовой работы в формате Microsoft Power Point 2010 в печатном (в виде приложения) и в электронном виде (на диске в виде файла pptx).

Имя файла презентации так же как должно иметь вид:

№группы_ФамилияИО_предмет_ККР№темы.pptx.

Например:

ЭБ_161_ИвановАС_ИСЭ_ККР56.pptx.

Презентация должна содержать 10-12 слайдов, первый из которых должен иметь вид титульного листа.

В работе обязательно использование скриншотов.

При выполнении работы необходимо выполнение следующих требований:

- при использовании литературных источников, нормативных и законодательных актов, статистических материалов в тексте необходимо делать соответствующие ссылки (с указанием источника заимствования и страниц), придерживаясь следующей рекомендации:

ссылки на используемый источник могут быть:

- 1) постраничными: в виде сноски в конце страницы;
- 2) в тексте, оформленные в квадратных скобках, на список литературы, приведенный в конце работы, где указывается порядковый номер источника в списке литературы и страница, на которую ссылается автор;

- в списке используемой литературы приводятся все источники с указанием выходных данных: автор, издательство, год издания, общее количество страниц, а при использовании периодического издания – автор, наименование издания, год, номер издания, страница, где статья помещается;

- в работе возможно приведение статического, графического материала, который оформляется в виде графиков, таблиц, схем и представляется или непосредственно в основной части работы или в приложении.

- при описании объекта исследования студент должен использовать дополнительные информационные источники, кроме рекомендованной учебной литературы: научные публикации, статьи, монографии, статистический материал, интернет-ресурсы.

Студенты в течение семестра освоили основные приемы работы с системой управления базами данных Microsoft MS Access. Однако, освоение возможностей СУБД было бы неполным без разработки целостного проекта, хотя бы средней сложности. Здесь под проектом средней сложности понимается база данных, структуру таблиц которой Вы должны разработать самостоятельно, и набор инструментов пользовательского интерфейса (запросов, отчетов и форм) для работы с информацией, хранящейся в базе данных.

При защите курсовой работы будут учитываться следующие его характеристики:

1. Оформление отчета должно соответствовать методическим указаниям кафедры по оформлению курсовых работ и дипломному проектированию. Отчет должен быть правильно скомпонован и красиво распечатан.

2. Курсовая работа принимается к защите только при наличии базы данных в электронном виде. Отчет и представленная БД должны находиться в строгом соответствии друг другу.

3. Отчет должен обязательно содержать перечисленные ниже разделы.

Введение. Примерно на одной-двух страницах следует изложить необходимость использования баз данных для автоматизации деятельности в той или иной области, полезность навыков работы с СУБД, ее возможности, а также назначение разрабатываемой базы данных.

1. Постановка задачи. Требуется четко изложить все требования к БД, которая будет создана. Типовыми ошибками является неполное перечисление требований или разработка возможностей, которые не описываются в постановке.

В подразделе «Требования к функциональным характеристикам» следует указывать требования к составу выполняемых функций, организации входных и выходных данных, требования к интерфейсу и т. п.

2. Проектирование структуры базы данных. В разделе приводится описание процесса проектирования БД: определение сущностей, ER- диаграмма, построение предварительных отношений, применение правил нормализации, таблица окончательных отношений. Эта работа проводится согласно данным методическим указаниям по проектированию баз данных реляционного типа.

3. Создание БД в MS Access. Описывается процесс создания таблиц базы данных, установления связей между ними, установления свойств каждой связи. Описывается информация, которая храниться в таблицах, типы данных, порядок и правила ввода данных.

4. Разработка запросов и пользовательского интерфейса. Приводится описание и распечатка всех разработанных запросов, отчетов и форм.

Заключение. Главная мысль этого раздела состоит в том, что проектирование выполнено и работоспособная БД создана.

5. ЗАЩИТА КУРСОВОЙ РАБОТЫ

Студенты представляют курсовые работы на кафедру в установленный срок (не позднее чем за три недели до окончания текущего семестра). По итогам проверки курсовой работы руководитель пишет на нее краткий отзыв. Положительный отзыв дает право на защиту курсовой работы. Суть защиты курсовой работы в основном сводится к обоснованию предложений, сформулированных студентом по рассматриваемой проблеме. Во время защиты студент должен ответить и на все замечания, сделанные руководителем как в отзыве, так и в тексте курсовой работы. Позднее представление курсовой работы влечет за собой задержку ее рассмотрения и снижение оценки за ее выполнение.

6. ПОРЯДОК РАЗРАБОТКИ РАЗДЕЛОВ КУРСОВОЙ РАБОТЫ *ПОСТАНОВКА ЗАДАЧИ*

Традиционно разработка любого программного обеспечения начинается с четкого описания выполняемых им функций - технического задания (ТЗ) на разработку. Базы данных не являются исключением, и поэтому основным содержанием первого раздела будет являться разработанное Вами техническое задание.

В качестве примера правил разработки технического задания можно привести ГОСТ 19.201-78, который регламентирует перечень и содержание разделов ТЗ. Однако, с учетом специфики курсовой работы, допустимы отступления от ГОСТ.

ТЗ на создание БД в процессе курсового проектирования должно включать **Титульный лист** и следующие перечисленные ниже пункты.

6.1. Цель разработки базы данных

Внимание! В первую очередь необходимо в самом общем виде определить область, для которой будет разрабатываться база данных. Это Вы должны сделать самостоятельно!

В качестве области использования БД можно взять гипотетическое или реальное предприятие, организацию или подразделение организации: оптовую базу; поликлинику; пункт автосервиса; клуб собаководства; учебную группу; кафедру и т.п. Возможна разработка БД для личных нужд: ведение электронного каталога аудио- и видеодисков; контроль расходов; хранение данных о книгах, полученных в библиотеке и т.д.

Внимание! Каждая курсовая работа строго индивидуальна. Во избежание повторений, тему КП необходимо согласовать с преподавателем.

Определившись с темой разработки необходимо описать причины, по которым разработка соответствующей БД является необходимой и полезной.

Пример. В курсовом проекте будет разработана БД для клуба собаководов «Дружок», предназначенная для хранения и обработки информации о членах клуба - владельцах собак, их питомцах и результатах участия собак в проходивших выставках.

В указанной организации состоит большое число собаководов-любителей, некоторые из которых имеют по несколько собак различных пород. Руководство клуба должно владеть достоверной информацией о всех его членах, иметь возможность своевременно оповещать их о намеченных мероприятиях и вести учет результатов участия собак в проходивших выставках. Без использования базы данных велика вероятность потери части необходимых сведений. Использование базы данных о членах клуба позволит уменьшить число

бумажных документов и значительно упростит работу сотрудников организации.

6.2. Требования к БД

1. Состав хранимой в БД информации.

В данном пункте необходимо определить основные объекты рассматриваемой предметной области, информация о которых должна содержаться в базе данных, и состав этой информации. Возможно, перечисление объектов Вы уже сделали в разделе «Цель разработки базы данных», тогда остается только детализировать информацию.

Минимальное количество основных объектов (*сущностей*), сведения о которых должны храниться в базе данных, определяется учебным планом по курсовому проектированию баз данных.

При выборе объектов Вы должны предусмотреть взаимосвязь между ними, это является одним из обязательных требований.

Пример.

БД должна содержать информацию следующего вида:

А. Информация о членах клуба:

- ✓ фамилия, имя, отчество;
- ✓ идентификационный номер;
- ✓ адрес;
- ✓ домашний телефон
- ✓ мобильный телефон;
- ✓ дата рождения.

Б. Сведения о собаках:

- ✓ кличка;
- ✓ номер документа
- ✓ порода;
- ✓ дата рождения.

В. Данные о результатах участия собак в выставках:

- ✓ название выставки;

- ✓ номер записи;
- ✓ дата проведения;
- ✓ места, занятые собаками на выставке.

Важно! Ключевые поля (свойства) должны быть подчеркнуты. Без указания ключевых полей дальнейшая разработка БД невозможна.

2. Выходная информация.

Это содержание запросов для получения полезной информации из базы данных. Минимальное число запросов устанавливается на усмотрение преподавателя, но не может быть менее 6. Рекомендуемое число запросов 10 и 6, для дневной и заочной форм обучения, соответственно. В числе запросов обязательно должны присутствовать запросы следующих типов:

- ✓ запросы на выборку с расчетом, выводящие информацию по одному из объектов предметной области;
- ✓ запросы на выборку, выводящие информацию по нескольким объектам предметной области - запросы на основе связанных таблиц;
- ✓ запросы на выборку с группировкой;
- ✓ перекрестные запросы;
- ✓ запрос-объединение;
- ✓ запросы действия - здесь необходимо разработать, по крайней мере, по одному запросу на обновление, добавление, удаление, создания таблиц, несколько DDL запросов и т.д.

3 Отчеты. По крайней мере, один из них должен содержать группировку строк и итоговые расчеты.

4. Требования к пользовательскому интерфейсу для работы с базой данных.

В данном пункте ТЗ Вы должны предусмотреть необходимость разработки форм для ввода и корректировки информации в базе данных, а также разработку кнопочной формы для запуска всех созданных запросов, отчетов и форм.

6.3. Требования к надежности

В этом разделе ТЗ необходимо описать, что Вы предпримете для обеспечения целостности и непротиворечивости информации в БД. Достаточно указать те средства MS Access, которые были изучены при выполнении лабораторных работ.

Внимание! Начинать разработку БД следует только после согласования ТЗ с преподавателем.

Вариант ТЗ для согласования темы может содержать краткую формулировку цели проектирования. Задание должно быть оформлено в виде распечатки, содержащей подпись работчика (студента), которая сдается для проверки. После утверждения ТЗ используется как основа для первого раздела курсовой работы и дополняется теоретическими аспектами.

Внимание! Изменение темы подписанного преподавателем ТЗ курсовой работы не допускается.

Проектирование структуры базы данных

1. Ошибочно было бы думать, что задача правильной организации данных связана с изучением какой-либо конкретной СУБД. В общей постановке она даже не связана с компьютеризацией. Уже много веков заполняются различные учетные книги, журналы, ведомости, списки. Сложилась и определенная правила, по которым определяется, какие данные должны в них входить.

Проектирование БД начинается с построения диаграмм ER-типа (Entity-Relation). Их русское название - диаграммы «сущность-связь». ER- диаграммы отражают структуру информации.

Внимательно изучите материалы лекций и раздел 3 данного пособия!!!

На основе ТЗ строится предварительная и уточненная ER-диаграммы.

Рекомендуемое число количество сущностей («прямоугольников») 6-9 и 4-5, для дневной и заочной форм обучения, соответственно.

Таким образом, на первом шаге проектирования создается ER- диаграмма. Для этого должна быть определены все сущности, все связи между ними, все классы принадлежности сущностей и все степени связи. В итоге информация, которая будет храниться в БД, будет структурирована.

Рисунки ER-диаграмм формируются в MS Office Visio. Для отчета по курсовому проектированию рисунки сохраняются в виде рисунков векторной графики и импортируются в документ MS Word.

2. Следующий шаг - формирование структуры данных для хранения информации. Такие структуры получили название отношений.

В соответствии с Правилами построения отношений на основе ER- диаграммы, формируется таблица предварительных отношений.

3. После получения предварительных отношений (в них содержатся пока только ключевые атрибуты) полезно подготовить список всех представляющих интерес атрибутов, которые не входят в ключи сущностей и назначить каждый из этих атрибутов одному из предварительных отношений.

4. Проектирование БД тесно связано с понятием **НОРМАЛЬНОЙ ФОРМЫ ОТНОШЕНИЯ**. Это означает, что проектируемые отношения должно удовлетворять определенным условиям (см. раздел 3).

5. После описания процессов нормализации базы данных, дается описание всех отношений, подлежащих реализации в СУБД MS Access, а также связей между ними. В соответст-

вующей таблице (*Таблице окончательных отношений*) приводятся названия отношений, перечень ключевых и неключевых атрибутов с их названием и, перечень атрибутов, участвующих в образовании связей между таблицами.

На этом проектирование структуры баз данных для решения задачи можно считать законченным.

6.4. Создание БД в MS Access

Далее, по полученным отношениям строятся таблицы базы данных. Их количество и перечень полей определяются из окончательного списка отношений. Характеристики полей задаются в соответствии с логикой и форматом хранимых данных.

При создании схемы данных важно обеспечить целостность связей. Для обеспечения целостности необходимо проследить за тем, чтобы связанные поля в главной и подчиненной таблицах имели одинаковые типы полей, например, табельные номера были бы везде текстовыми и т.д. Требования к целостности данных подразумевают очевидные допустимые значения полей: поле для связи в подчиненной таблице не может содержать значения, которые не описаны в главной, а также пустых значений.

Совет. Для создания правильной схемы данных необходимо заполнить в таблицах несколько строк.

После создания схемы данных и заполнения таблиц можно приступить к разработке запросов. Запросы, отчеты и формы разрабатываются в соответствии с техническим заданием.

Запросы можно создавать в конструкторе.

Важно. Минимальным требованием к пользовательскому интерфейсу, является подключение всех созданных запросов к форме.

Кнопки, позволяющие запускать запросы, должны иметь понятное описание, что за запрос будет запускаться. Это легко сделать, используя на форме поясняющий текст – элементы управления с текстовой строкой.

6.5. Оформление отчета

Главным итогом проделанной работы является отчет, составленный по правилам оформления курсовых работ и распечатанный в формате А5 с двух сторон. Схема рис. 61 показывает структуру и последовательность проектирования, а также состав материалов, которые должны войти в итоговый отчет.

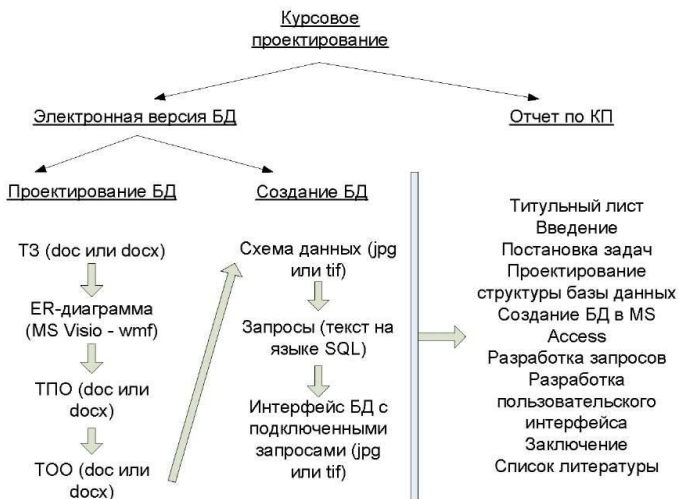


Рис. 6.1. Последовательность проектирования БД

В отчете описывается процесс создания таблиц базы данных, установления связей между ними, установления свойств каждой связи. Описывается информация, которая храниться в таблицах, используемые типы данных, ограничения на размер полей, порядок и правила ввода данных. Можно привести заполненные таблицы (наиболее интересные с точки

зрения разработчика), а также их структуру в режиме Конструктора.

Особое внимание надо уделить качеству оформлению рисунков. Необходимо грамотно организовать подписи к рисункам и ссылки на рисунки из текста отчета. Правила оформления рисунков и таблиц содержатся, например, в методических указаниях по дипломному проектированию.

В отчете обязательно приводится рисунок схемы данных. Для снятия копии экрана используется сочетание клавиш Alt-PrtSc или просто PrtSc. Копия экрана помещается в любой графический редактор, где вырезается нужная область со схемой данных.

Для каждого запроса приводится текст запроса на языке SQL, смысл запроса, описание. Можно привести рисунок, иллюстрирующий результат выполнения запроса.

В отчете по курсовой работе приводятся внешний вид отчетов.

Обязательно приводятся описание интерфейса и рисунки, показывающие возможность запуска разработанных запросов.

Для защиты проделанной работы по проектированию БД, каждому студенту необходимо представить: электронную версию БД - приложение для СУБД MS MS Access и Отчет по курсовому проектированию (курсовой работе).

7. ОСНОВНЫЕ ПРАВИЛА ПРОЕКТИРОВАНИЯ БД

7.1. Правила построения ER-диаграмм

Исходные данные для проектирования

Пусть проектирование БД ведется для гипотетического малого предприятия «МакроСофт», руководство которого решило внедрить информационную систему на основе СУБД.

Пусть на предприятии работает около 10 человек. Предприятие продает по заказам свои разработки ПО, а также оборудование (компьютеры различных конфигураций, собираемые из комплектующих) и комплектующие (мыши, платы памяти, дисководы, дискеты).

Сотрудники работают по несколько человек в комнатах, арендуемых в НИИ. В каждой комнате установлен отдельный телефон. Генеральному директору этого предприятия необходимо иметь сведения о номерах комнат, рабочих и домашних телефонах работников, их домашних адресах. Кроме того, он предполагает планировать работу каждого из своих сотрудников. Наконец, ему нужны списки заказов, которые выполняет его предприятие, и цены, по которым поставщики продают комплектующие изделия.

Информационная система должна выводить следующие справки:

- ✓ анкетные данные всех работников;
- ✓ номер рабочего телефона каждого сотрудника;
- ✓ список ответственных за техническое состояние принадлежащих предприятию компьютеров и противопожарное состояние комнат;
- ✓ какие работы по заказам, в какие сроки и по каким разработкам выполняют работники;
- ✓ кем и на какую сумму сделан заказ, что он включает; какие поставщики (наименование фирмы, адрес) участвуют в

поставке комплектующих на данную конфигурацию оборудования;

✓ по каким ценам заказчики покупают оборудование и комплектующие.

Основные определения

СУЩНОСТЬ - некоторый объект, представляющий интерес для организации. Сущность должна иметь **ЭКЗЕМПЛЯРЫ**, отличающиеся друг от друга и допускающие однозначную идентификацию *Единственный определяющий признак сущности - это, как правило, сущестительное.*

ПРИМЕРЫ:

РАБОТНИКИ. РАБОТЫ. ДЕТАЛИ. СТУДЕНТЫ. ЭКЗАМЕНЫ

Внимание! Наиболее распространенная ошибка проектирования заключается в том, что в набор сущностей включается и сама фирма - Заказчик. Она действительно является объектом, представляющим большой интерес, но не имеет экземпляров, поэтому сущность для нее выделять не следует.

АТТРИБУТ - свойство сущности

ПРИМЕР:

Сущность РАБОТНИКИ имеет атрибуты: НомерТелефона, Фамилия, Адрес.

Разница между сущностью и атрибутом заключается в том, что экземпляры сущностей (работники), согласно определению, представляют интерес сами по себе. Атрибуты же относятся к сущности и самостоятельного интереса не представляют (что такое просто номер телефона и зачем нужен неизвестно чей адрес?).

Теперь должно быть ясно, для чего выделяются сущности. Данные об их экземплярах будет удобно представлять в

табличной форме, так как все экземпляры однотипны и имеют одинаковые наборы атрибутов. Выделение сущностей - первый шаг проектирования. Он помогает декомпозировать задачу. Часть работы можно будет сделать отдельно для каждой сущности.

Задача выделения сущностей среди набора данных не проста и даже не всегда однозначна. Общая рекомендация здесь - не торопиться вводить дополнительные сущности. Как будет ясно из примера проектирования, новую сущность можно легко добавить и позже.

КЛЮЧ СУЩНОСТИ - атрибут или набор атрибутов, используемый для идентификации экземпляра сущности

Это определение вводится для того, чтобы обеспечить провозглашенное требование однозначной идентификации экземпляров сущностей. У нескольких работников фирмы может быть одинаковая фамилия. Некоторые из них могут быть родственниками и жить в одной квартире. Для того, чтобы различать работников предприятия, им присваиваются табельные номера или выдаются пропуска¹.

Общим для ключа сущности и всех других ключей, рассматриваемых ниже, является то, что в **качестве ключа следует использовать минимальный набор атрибутов**, так, чтобы при удалении одного из них этот набор перестал бы быть ключом сущности. Например, использование для сущности РАБОТНИКИ и фамилии и табельного номера неправильно, так как идентификацию можно осуществить только по табельному номеру.

Если определена сущность ВЫПЛАТЫ, то ее экземпляры отличаются тем, когда (атрибут Дата), кому (ТабельныйНомер) и за что (ВидПлатежа) они производились. Ключ будет включать одновременный набор этих атрибутов. Если

¹ Студенты высшего учебного заведения имеют уникальный номер зачетной книжки. Посетители библиотеки регистрируются по номеру читательского билета. Оборудование снабжается инвентарными номерами.

удалить из него, например, атрибут ТабельныйНомер, то не будет обеспечиваться однозначная идентификация, так как в один день один вид выплат могли получить несколько работников.

ЗАМЕЧАНИЕ. Для удобства работы ключи принято подчеркивать. Лучше привыкнуть к этому сразу.

СВЯЗЬ - соединение между двумя или более сущностями

Связь между двумя сущностями называется **БИНАРНОЙ**. Это наиболее простой и широко распространенный тип связи. Связь между тремя сущностями называется **ТЕРНАРНОЙ**. В общем случае связи могут быть **П-АРНЫМИ**.

Единственный определяющий признак связи - это, как правило, глагол.

ПРИМЕРЫ:

РАБОТНИКИ—ВЫПОЛНЯЮТ—РАБОТЫ.

СТУДЕНТЫ—СДАЮТ—ЭКЗАМЕНЫ.

Связи отражают взаимосвязанность хранящихся в БД данных. Без учета связей невозможно будет построить запросы, включающие данные из нескольких таблиц. С помощью связей можно определить, какие работники выполняют каждую работу и какие работы выполняются каждым работником; какие студенты сдают каждый из экзаменов и какие экзамены сдаются каждым студентом.

Именно введение связей делает хранение данных в БД столь гибким и универсальным².

СТЕПЕНЬ СВЯЗИ. Что такое степень связи, проще всего пояснить на примере.

² Хранение информации о связях всегда было одной из главных проблем в СУБД. Они строились на различных моделях данных, главное отличие между которыми заключалось в способе отражения связей. Известны, в частности, иерархические, сетевые и реляционные модели данных. В первых двух для связей использовались специальные структуры данных. В реляционной модели данных информация о сущностях и связях хранится в едином формате реляционных таблиц, что обусловило широкое ее распространение.

*ПРИМЕР. Для связи
РАБОТНИКИ—ВЫПОЛНЯЮТ—РАБОТЫ*

Возможны следующие степени связи:

1:1 - один работник выполняет одну работу;

1:n - один работник выполняет несколько работ, но каждая работа выполняется одним работником;

n:1 - несколько работников выполняют одну работу, но каждый работник выполняет только одну работу;

n:m - один работник выполняет несколько работ и одна работа выполняется несколькими работниками.

КЛАСС ПРИНАДЛЕЖНОСТИ СУЩНОСТИ. Возможны следующие классы принадлежности: **ОБЯЗАТЕЛЬНЫЙ** и **НЕОБЯЗАТЕЛЬНЫЙ**.

Если нет, и никогда не может быть работника, который не выполняет ни одной работы, то класс принадлежности сущности РАБОТНИКИ является обязательным. В фирме «МакроСофт» часть работников может и не заниматься договорами. Тогда класс принадлежности сущности РАБОТНИКИ - необязательный.

Иллюстрация новых понятий дана на рис. 7.1. Сущность РАБОТНИКИ имеет четыре экземпляра, а РАБОТЫ - пять. Имеется четыре экземпляра связей, показывающих, кто какую работу выполняет. Из рисунка следует, что класс принадлежности является необязательным для работ, так как имеются работы, которые никем не выполняются, и обязательным для работников, так как все работники заняты на различных работах. Степень связи - **n** (для работников) : **1** (для работ).

ПРАВИЛО ОПРЕДЕЛЕНИЯ СТЕПЕНИ СВЯЗИ: одну работу могут делать **НЕСКОЛЬКО РАБОТНИКОВ** ($n : \dots$), но один работник может делать **ОДНУ РАБОТУ** ($\dots : 1$).

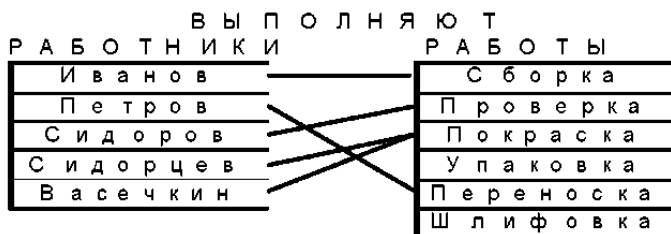


Рис. 7.1. Связи между Работниками и Работами

Степень связи и класс принадлежности сущностей определяются фундаментальными свойствами тех объектов, данные о которых предполагается хранить в БД, поэтому к определению этих характеристик следует относиться очень внимательно. Выводы, сделанные по рис. 3.1, являются только предварительными и требуют уточнения у Заказчика.

Ошибка, допущенная в этом вопросе, скорее всего приведет впоследствии к значительным и дорогостоящим переделкам БД и программного обеспечения.

ПРИМЕР. Для малого предприятия «МакроСофт» важно установить, какая информация будет храниться в создаваемых БД: будут ли в таблицу РАБОТНИКИ занесены все сотрудники или только работающие по договорной тематике; будут ли учитываться все работы или только работы в рамках договоров; может ли один работник выполнять несколько работ; сохраняются ли сведения об уже выполненных работах; записываются ли в БД работы, для которых еще не выбран исполнитель и т. п.

Графическое изображение сущностей, связей и ключей называется ER-диаграммой.

ПРИМЕР. Рис.7.2 отображает ER-диаграмму для рис.7.1. Ключом сущности РАБОТНИКИ является табельный номер работника, обозначаемый НТаб, а ключом сущности РАБОТЫ - номер работы НРаб.

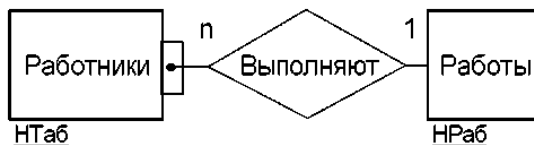


Рис. 7.2. ER-диаграмма для рис.7.1

При построении ER-диаграмм используются следующие правила.

Сущности обозначаются прямоугольниками.

Связи обозначаются ромбами, соединенными линиями (без стрелок!) с соответствующими сущностями.

Названия сущностей и связей пишутся внутри обозначающих их элементов.

Обязательность связи (слева на рис.3.2) обозначается небольшим прямоугольником; иногда в его центре рисуется жирная точка, или просто жирной точкой.

Необязательность связи (справа на рис.3.2) не имеет дополнительных обозначений.

Степень связи указывается около сущностей, которые она связывает.

Ключи сущностей располагаются около значков сущностей.

Ключи связей не отображаются.

Итак, первый шаг проектирования - построение ER-диаграмм. В общем случае каждая сущность может быть связана с несколькими другими. Для каждой связи требуется

определить класс принадлежности каждой из сущностей и степени каждой связи.

Пример построения ER-диаграмм

На основании собранных исходных данных Разработчик информационной системы для фирмы «МакроСофт» может сформировать сущности, то есть определить список объектов, представляющих интерес для Заказчика. Это: ЗАКАЗЫ, РАБОТНИКИ, РАЗРАБОТКИ, КОМПЬЮТЕРЫ, КОМПЛЕКТУЮЩИЕ, ПОСТАВЩИКИ. Рис.7.3 - первая попытка сформировать связи между этими сущностями на основе имеющейся информации, опыта и «здравого смысла».

Естественными в данном случае будут следующие допущения, касающиеся степеней связи и классов принадлежности сущностей:

- ✓ один заказ может иметь компьютеры различных конфигураций и одна конфигурация компьютера может присутствовать в различных заказах;

- ✓ то же относится и к комплектующим;

- ✓ в состав компьютера обязательно входит несколько типов комплектующих (иначе просто не собрать компьютер), однако есть комплектующие, не входящие ни в одну конфигурацию (например, дискеты);

- ✓ одни и те же комплектующие могут входить в различные конфигурации;

- ✓ каждый поставщик обязательно что-то поставляет, иногда - несколько типов комплектующих (интересно хранить сведения только о фирмах, поставляющих что-либо);

- ✓ все комплектующие кем-то поставляются, может быть, несколькими поставщиками (сама фирма не занимается изготовлением комплектующих).

Все это отражено на рис 7.3.

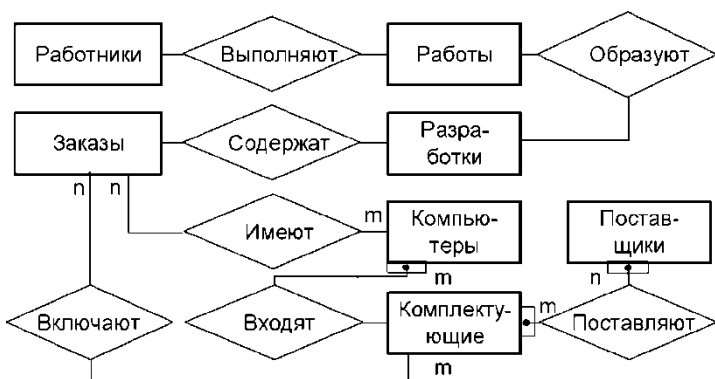


Рис. 7.3. Исходный вариант ER-диаграммы

Оставшиеся неясными классы принадлежности и степени связи необходимо выяснить у Заказчика. Для этого надо сформировать вопросы к нему. Вот список возможных вопросов для второго разговора с Заказчиком. Здесь же в скобках даны ответы Заказчика. Следует, однако, помнить, что эти ответы характеризуют **только конкретное предприятие** «МакроСофт».

Верны ли сделанные допущения? (Да.)

Как связаны РАБОТНИКИ и РАБОТЫ:

- а) каждый делает только одну свою работу;
- б) каждый делает несколько работ;
- в) каждая работа делается несколькими работниками;
- г) несколько работников участвуют в нескольких работах.

Все ли работники заняты выполнением работ по договорам? (Нет.)

Но нужно хранить сведения обо всех работниках? (Да.)

Всегда ли известно, кто выполняет или будет выполнять каждую из работ? (Да.)

Действительно ли каждая разработка делится на несколько работ, причем эти работы для каждой разработки свои? (Да.)

Может ли заказ иметь несколько разработок? (Нет.)

Можно ли одну разработку ПО включить в несколько заказов или каждая разработка уникальна? (Уникальна. Даже если поставляется имеющееся программное обеспечение, то его следует настроить на конкретное применение).

Существуют ли заказы без разработки программного обеспечения, только на компьютеры и комплектующие? (Нет.)

Планируете ли Вы разработки без заказов, на перспективу? (Нет.)

Далее Заказчик уточнил, что ему хотелось бы иметь сведения не только о фирмах, сделавших заказы, но и о потенциальных клиентах для рассылки рекламных проспектов. Поэтому для каждого клиента предприятия, как сделавшего заказ, так и пока не имеющего заказа, требуется знать название фирмы и адрес. На основе состоявшегося разговора была составлена уточненная ER-диаграмма, в которую введены также ключи сущностей (рис.7.4). Имена³ и содержание ключевых атрибутов сводятся в таблицу (табл.7.1).

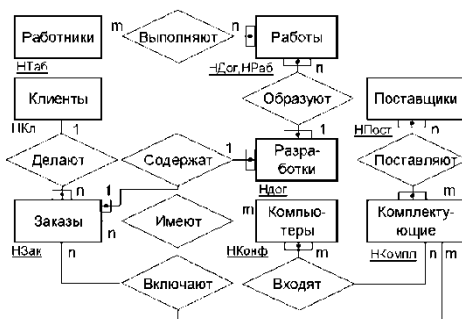


Рис. 7.4. Уточненная ER-диаграмма

³ Хотя имена в БД назначаются произвольно, лучше разработать для себя определенные правила их формирования. Тогда работа сильно облегчается, так как имя будет нести информацию о содержании данных. Здесь применено следующее правило: если атрибут является номером, то его первая буква - Н. Далее идет сокращенное название (Таб - табельный, Комп - компьютер и т.д.). Если имя состоит из нескольких слов, то каждое слово начинается с заглавной буквы и записывается без пробела.

Таблица 7.1

Имена и содержание ключевых атрибутов

Имя	Содержание
1	2
НДог	Номер договора на разработку. Обычно ведется учет договоров и им присваиваются уникальные номера. Если договор составляет клиент, может оказаться, что у двух разных клиентов имеются одинаковые номера договоров. Но в ЗАО «МакроСофт» каждому договору присваивается свой учетный номер.
НЗак	Номер заказа. Этот номер также присваивается ЗАО «МакроСофт»;
НКл	Номер клиента. Названия фирм могут повторяться. Кроме того, если Заказчик - крупная фирма, то заказы могут делать различные ее отделы. ЗАО «МакроСофт» ведет учет своих клиентов, присваивая им уникальные номера.
НКомпл	Номер вида комплектующего изделия. Штрих-коды, широко распространенные в настоящее время, уникально идентифицируют каждый вид товара. Это представление достаточно компактно (обычно содержит 13 цифр). С помощью штрих-кодов легко вести автоматизированный учет движения комплектующих изделий внутри фирмы. Поэтому в данном случае используется штрих-код в цифровом выражении.
НКонф	Номер конфигурации поставляемого компьютера. Таких конфигураций имеется порядка 20. Они пронумерованы ЗАО «МакроСофт».
НПост	Номер поставщика. Задается аналогично номеру клиента.

1	2
НРаб	Номер работы в рамках договора. В рамках одного договора может быть несколько работ. Это сборка компьютера, тестирование, программирование и проч. В ЗАО «МакроСофт» работы нумеруются по порядку их выполнения, причем эта нумерация начинается заново для каждого договора. Поэтому много договоров имеют работу №1, №2 и т. д. Для идентификации конкретной работы требуется знать номер договора и номер работы по этому договору
НТаб	Табельный номер работника. Присвоением табельных номеров каждому своему работнику занимается каждая фирма. Чаще всего этот номер содержит только цифры, но иногда в него вводятся дополнительные буквенные обозначения.

Следующий шаг - формирование структур данных для хранения информации. Такие структуры получили название отношений.

7.2. Правила получения предварительных отношений

Основные определения

Данные в БД представляются в виде отношений. Теория отношений достаточно формализована, так что перед началом изложения материала необходим ряд определений.

МНОЖЕСТВО - совокупность произвольных объектов, объединенных по некоторому признаку.

Каждый объект называется **ЭЛЕМЕНТОМ МНОЖЕСТВА**.

Элементы не упорядочены и не могут дублироваться.

ОТНОШЕНИЕ R есть множество упорядоченных n -КОРТЕЖЕЙ вида: $\langle d_1, d_2, \dots, d_n \rangle$,

где d_i - элемент множества D_i ; d_2 - элемент множества D_2 ; d_n - элемент множества D_n .

$D_i, i = 1, 2, \dots, n$ называется ДОМЕНОМ отношения R .

При проектировании БД элементами множеств являются объекты реального мира, которые описываются при помощи системы характеристик, называемых АТРИБУТАМИ. Значение характеристики называется ЗНАЧЕНИЕМ АТТРИБУТА.

В состав ОПИСАНИЯ ОТНОШЕНИЯ входят:

ЗАГОЛОВОК ОТНОШЕНИЯ - множество атрибутов, описывающих объект, причем каждый атрибут соответствует некоторому домену; ТЕЛО ОТНОШЕНИЯ - множество кортежей.

ПРИМЕР. Рисунок 7.5 представляет собой пример отношения.

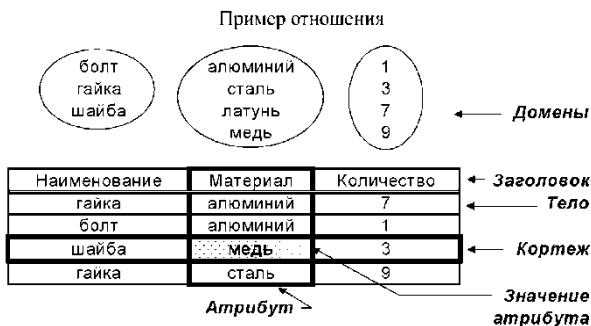


Рис. 7.5. Пример отношений

Данные, содержащиеся обычно в традиционных, «ручных» таблицах достаточно произвольного вида, сначала организуются в отношения - определенные выше структуры табличного вида. Тем самым они становятся пригодными для хранения в таблицах реляционных БД. Отношения определяют состав и структуру таблиц БД, в которых будут храниться.

Полезно проследить соответствие терминов, используемых для таблиц, в теории отношений и СУБД. Это сделано в табл.7.2.

Таблица 7.2

Соответствие терминов при переходе к таблицам БД

Таблицы	Теория отношений	Таблицы базы данных
Таблица Строка Столбец Клетка	Отношение Кортеж Атрибут Значение атрибута	Таблица (ранее - файл БД) Запись Поле Элемент данных

Еще один термин - потенциальный ключ отношения - имеет в теории отношений тот же смысл, что и ключ сущности в ER-диаграммах.

ПОТЕНЦИАЛЬНЫЙ КЛЮЧ ОТНОШЕНИЯ - атрибут или набор атрибутов, который может быть использован для однозначной идентификации кортежа. Потенциальный ключ не должен иметь дополнительных атрибутов.

Если из первичного ключа убрать хотя бы один атрибут, то он уже не будет первичным ключом.

Потенциальный ключ, состоящий из одного атрибута, называется простым. Потенциальный ключ, состоящий из нескольких атрибутов, называется составным.

Существуют правила перехода от ER-диаграмм к набору отношений.

Правила построения предварительных отношений по ER- диаграммам

Следующий шаг проектирования структуры данных заключается в получении предварительных отношений и их предполагаемых первичных ключей. Для этого необходимо руководствоваться следующими **ШЕСТЬЮ ПРАВИЛАМИ ДЛЯ БИНАРНЫХ СВЯЗЕЙ** и **ОДНИМ** правилом для **N-арной связи**.

ПРАВИЛО 1. Если степень связи *1:1* и класс принадлежности обеих сущностей *обязательный*, то такая связь пре-

образуется в *ОДНО* отношение, первичным ключом которого становится ключ любой из сущностей.

ПРИМЕР. Рассмотрим связь РАБОТНИКИ — ВЫПОЛНЯЮТ— РАБОТЫ (рис. 7.6).

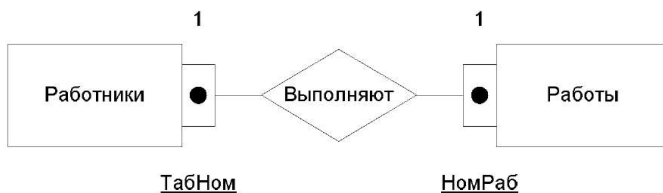


Рис. 7.6. Связь 1:1 с обязательными классами принадлежности

Пусть каждый работник обязательно выполняет одну и только одну работу, и нет работы, которую никто бы не выполнял, то есть число работ равно числу работников. Тогда для этой связи достаточно одного отношения, ключом которого может быть как ТабНом (табельный номер работника), так и НомРаб (номер работы). Если требуется хранить дополнительные атрибуты работников и работы, то они могут находиться в одном отношении, так как по ТабНом можно определить работника, а по НомРаб - выполняемую единственным работником работу; все работы и все работники содержатся в одном отношении (рис. 7.7).



Рис. 7.7. Создание отношения по Правилу 1

ПРАВИЛО 2. Если степень связи $1:1$ и класс принадлежности одной из сущностей - *обязательный*, а другой - *необязательный*, то формируются *ДВА* отношения, по одному для каждой сущности.

Ключ каждой сущности становится первичным ключом соответствующего отношения. Ключ сущности с *необязательным* классом принадлежности добавляется как атрибут в отношение для сущности с *обязательным* классом принадлежности.

ПРИМЕР. Пусть каждый из работников обязательно выполняет какую либо свою работу (класс принадлежности сущности РАБОТНИКИ *обязательный*), но есть одна или более работ, которые работники пока не выполняют (класс принадлежности сущности РАБОТЫ *необязательный*). Очевидно, что перечень работ содержит больше записей, чем число работников (рис. 7.8).

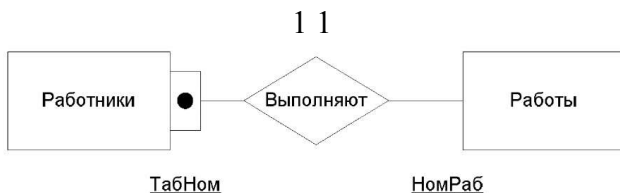


Рис. 7.8. Связь 1:1 с одним обязательным классом принадлежности

В этом случае формируется два отношения, причем поле НомРаб добавляется в отношении РАБОТНИКИ для связи (рис. 7.9).

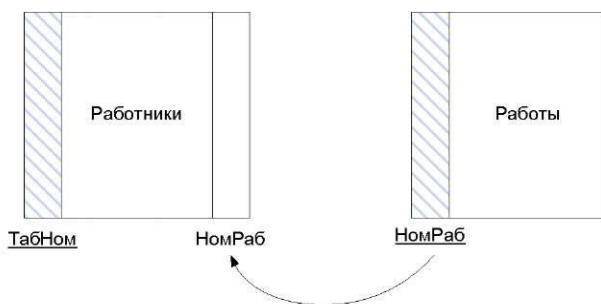


Рис. 7.9. Связь 1:1 с одним обязательным и с одним необязательным классами принадлежности

ПРАВИЛО 3. Если степень связи $1:1$ и класс принадлежности обеих сущностей *необязательный*, то необходимы *ТРИ* отношения: по одному для каждой сущности и одно для связи. Ключами первых двух отношений будут ключи сущностей, а ключом третьего - ключ любой из сущностей.

ПРИМЕР. Данную ситуацию иллюстрирует рис. 7.10. Не каждый работник выполняет работу, описанную в перечне работ, но и не все работы обязательно выполняются данными работниками.

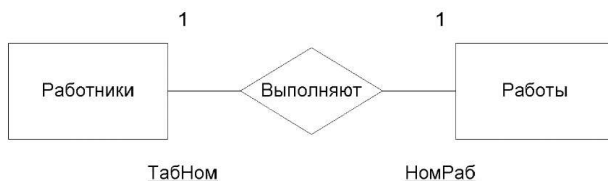


Рис. 7.10. Связь 1:1 с необязательными классами принадлежности

В отличие от правила 2, в таблицу работники уже нельзя добавить столбец с номерами работ, так как они будут иметь пустые значения.

Остается построить третью таблицу, ВЫПОЛНЯЕМЫЕ РАБОТЫ, в которой будут добавлены два столбца, содержащие работников и указание на выполняемые ими работы рис. 7.11.

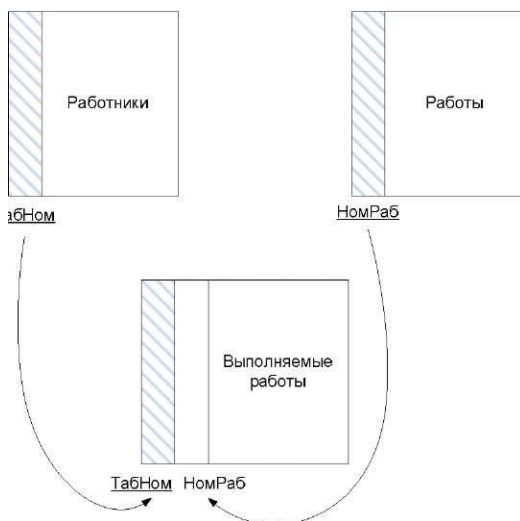


Рис. 7.11. Формирование таблицы для связи, согласно Правилу 3

ПРАВИЛО 4. Если степень связи $1:n$ и класс принадлежности n -связной сущности *обязательный*, то необходимы *ДВА* отношения, по одному для каждой сущности. Ключами этих отношений станут ключи каждой сущности. Ключ одно-связной сущности добавится как атрибут в отношение для n -связной сущности.

ПРИМЕР 1. Пусть степень связи РАБОТНИКИ — ВЫПОЛНЯЮТ — РАБОТЫ будет $1:n$ (рис. 7.12). Это означает, что один работник может выполнять несколько работ (степень связи n для работ), причем все работы обязательно кем-то выполняются (класс принадлежности сущности РАБОТЫ - обязательный).



Рис. 7.12. Степень связь 1:n с обязательным классом принадлежности для n-связанной сущности

Чтобы описать, кто из работников выполняет какую работу, достаточно в отношении РАБОТЫ добавить атрибут ТабНом, идентифицирующий работника (рис. 7.13).

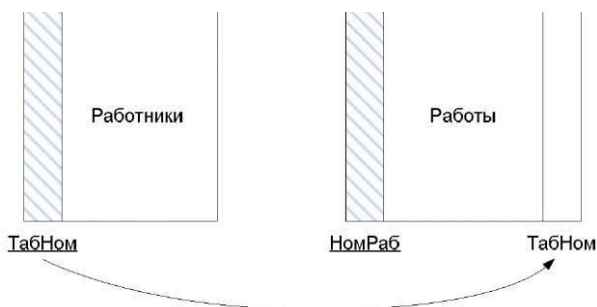


Рис. 7.13. Одна из наиболее часто встречающихся связей в БД

ПРАВИЛО 5. Если степень связи $1:n$ и класс принадлежности n-связанной сущности *необязательный*, то необходимы *ТРИ* отношения: по одному для каждой сущности и одно для связи. В отношении для связи среди атрибутов должны быть ключи каждой сущности. Ключами первых двух отношений будут ключи сущностей, а ключом третьего - ключ n-связанной сущности.

ПРИМЕР. Пусть сущность РАБОТНИКИ для связи РАБОТНИКИ — ВЫПОЛНЯЮТ — РАБОТЫ содержит сведения о всех работниках, сущность РАБОТЫ - обо всех работах (рис. 7.14). Правило применимо в том случае, если один работник выполняет несколько работ, но один работник в каждый момент времени занят только своей группой работой (степень связи **n** для работ и **1** для работников). Кроме того, не все работы выполняются работниками (класс принадлежности **n**-связной сущности необязательный). Не имеет значения, все ли работники заняты работой из списка работ (класс принадлежности сущности РАБОТНИКИ произволен).

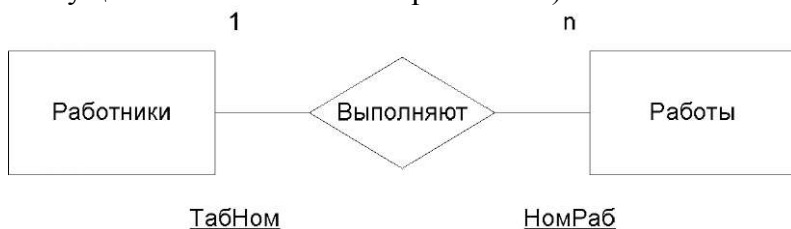


Рис. 7.14. Степень связь 1:n с необязательным классом принадлежности для n-связанной сущности

Чтобы указать, кто над чем работает, нельзя добавить атрибут работы в отношении РАБОТНИКИ, так как бывают ситуации, когда кто-то не выполняет никакой работы. Нельзя добавить и атрибут работника в отношении РАБОТЫ, так как существуют работы, которые не выполняются работниками из списка. Остается ввести новое отношение, ВЫПОЛНЯЕМЫЕ РАБОТЫ, с атрибутами ТабНом и НомРаб, в котором будет указано, кто над чем работает (рис. 7.15). Ключом отношения при этом становится НомРаб.

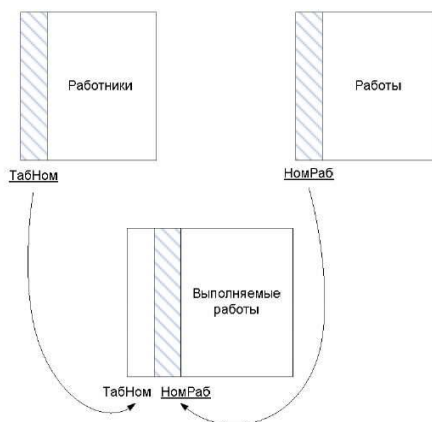


Рис. 7.15. Формирование таблицы для связи, согласно Правилу 58

ПРАВИЛО 6. Если степень связи $m:n$, то необходимы *ТРИ* отношения: по одному для каждой сущности и одно для связи. В отношении для связи среди атрибутов должны быть ключи каждой сущности. Ключами первых двух отношений становятся ключи сущностей, а ключом третьего - ключи обеих сущностей.

ПРИМЕР (рис. 7.16). Если для предыдущего примера допустить, что одну работу могут выполнять одновременно несколько работников, то все рассуждения будут почти аналогичны. Разница только в том, что в отношении **ВЫПОЛНЯЕМЫЕ РАБОТЫ** может встречаться несколько раз запись о работах. Поэтому ключ отношения **ВЫПОЛНЯЕМЫЕ РАБОТЫ** будет составным (рис. 7.17).



Рис. 7.16. Степень связь $n:m$

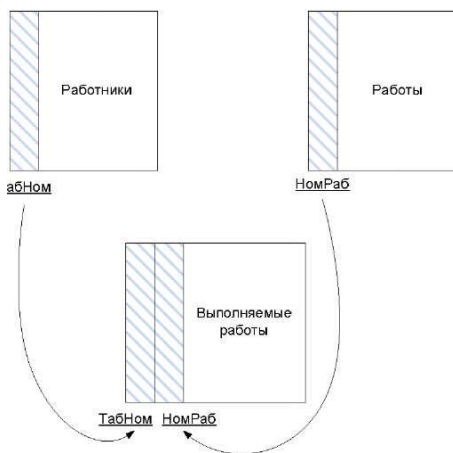


Рис. 7.17. Формирование таблицы для связи, согласно Правилу 7

Связь "многие-ко-многим" является наиболее универсальной и чаще всего встречается при описании связи между объектами.

Замечание. В ряде случаев, в таблице для связи полезно ввести свой собственный ключ (задать новое ключевое поле). Поля, которые входили до этого в составной ключ, становятся обычными полями для связи (неключевыми атрибутами).

Приведенные правила охватывают все возможные в бинарных связях ситуации.

n-АРНЫЕ СВЯЗИ. Иногда возникают случаи, когда связи между сущностями получаются сложными и не сводятся к бинарным.

ПРИМЕР. Пусть работники выполняют работы на основании заключенных контрактов (сделанных заказов, рис. 3.18). Здесь важно указать именно взаимосвязь нескольких событий, их одновременность: номер заказа, кто работал над конкретным заказом и какие работы выполнял. $m:n$

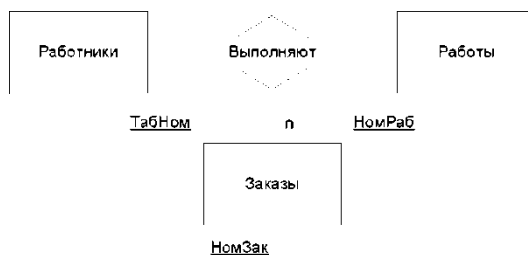


Рис. 7.18. ER-диаграмма с n-арной связью

Для **n**-арных связей существует одно общее правило.

ПРАВИЛО 7. Для **n**-арной связи ($n > 2$) требуется $n+1$ отношение: по одному для каждой сущности и одно для связи. Ключи каждой сущности превращаются в ключи соответствующих отношений. Кроме того, все ключи входят как атрибуты в отношение для связи. Ключ отношения для связи определяется по-разному, в зависимости от классов принадлежности и степеней связи.

На основании Правила 7 формируется сущность **ВЫПОЛНЯЕМЫЕ РАБОТЫ**, в составной ключа которой входят ТабНом, НомРаб и НомЗак. Связи будут иметь вид (рис. 7.19):

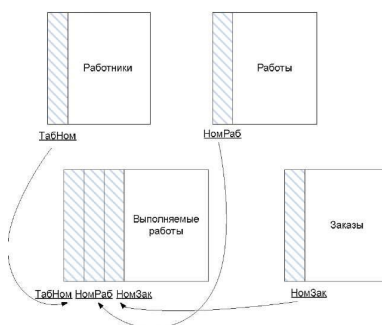


Рис. 7.19. Формирование таблицы для связи, согласно Правилу 7

Иногда целесообразно вместо составного ключа создать новый ключ отношения. Поля, входившие в состав составного ключа, остаются в качестве полей для связи.

Составление Таблицы предварительных отношений. В соответствии с вышеприведенными правилами, для проектируемой БД формируются Таблица предварительных отношений (например, табл. 7.4). Таблица содержат пока только ключевые атрибуты (подчеркнуты) и атрибуты, вводимые по правилам построения (указаны курсивом). В третьей колонке указывается номер правила, на основании которого было образовано данное отношение, либо когда в существующее отношение были внесены изменения (добавлено поле для связи).

Как показывает опыт, наиболее часто используются правила 4 и 6, значительно реже 7. Другие правила используются крайне редко, когда связи между объектами не удастся свести к правилам 4 или 6.

Далее проводится процесс распределение атрибутов по отношениям. Рекомендуется сделать копию табл. 7.3 и вместо многоточий вписать неключевые атрибуты, состав которых, был описан еще в техническом задании. Следует отметить, что большинство созданных отношений не нуждаются в корректировке. Однако для грамотного проектирования БД необходимо проверить отношения на предмет соответствия их той или иной Нормальной форме.

Таблица 7.3

Таблица предварительных отношений

Название отношения	Ключевые поля и поля для связи	Используемое правило
1	2	3
Работники	(ТабНом, ...)	-
Работы	(НомРаботы, <u>ТабНом</u> ,	4

1	2	3
Клиенты	(НомКлиента, ...)	-
Заказы	(НомЗаказа, ТабНом, НомКлиента, ...)	4, 4
Поставщики	(НомПоставщика, ...)	-
Продукция	(Артикул, ...)	-
Поставляют	(Артикул, НомПоставщика, ...)	6

7.3. Правила нормализаций отношений

Основные определения

Если отношение находится в нормальной форме, то снимаются многие проблемы хранения и обработки данных. Разрабатываемые методы проектирования применимы к определенным нормальным формам.

Известны три нормальные формы и формы высших порядков. Здесь будут рассмотрены первые три нормальных формы и нормальная форма Бойса-Кодда.

Нормальные формы строятся по следующему принципу: чтобы отношение находилось в некоторой нормальной форме, требуется, чтобы оно находилось в предыдущей нормальной форме и выполнялись определенные дополнительные условия. Исключением является только первая нормальная форма.

Отношение находится в первой нормальной форме (1НФ) тогда и только тогда, когда на пересечении каждого столбца и каждой строки находятся только элементарные значения атрибутов.

Первая нормальная форма - это обычное отношение. Согласно определению отношений, любое отношение автоматически уже находится в 1НФ. Иными словами, данные, представленные в виде двумерной таблицы, являются первой нормальной формой реляционной модели данных.

Отношение находится во второй нормальной форме (2НФ) тогда и только тогда, когда отношение находится в 1НФ, и нет, неключевых атрибутов, зависящих от части сложного ключа. (Неключевой атрибут - это атрибут, не входящий в состав никакого потенциального ключа).

Каждая таблица должна иметь уникальный идентификатор, или первичный ключ, который может состоять из одного или нескольких полей.

Замечание. Если потенциальный ключ отношения является простым, то отношение автоматически находится в 2НФ.

Определение. Функциональная зависимость $A \rightarrow B$ называется транзитивной (опосредованной), если существует набор атрибутов C такой, что:

- существует функциональная зависимость $A \rightarrow C$;
- существует функциональная зависимость $C \rightarrow B$.

Отношение находится в третьей нормальной форме (3НФ) тогда и только тогда, когда оно находится во второй нормальной форме и не содержит транзитивных зависимостей.

То есть в таком отношении все неключевые атрибуты зависят от ключа напрямую.

Для каждого значения первичного ключа значения в столбцах данных должны относиться к объекту таблицы и полностью его описывать.

Это правило используется двояко. Во-первых, в таблице не должно быть данных, не относящихся к объекту, определяемому первичным ключом. Во-вторых, данные в таблице должны полностью описывать объект.

Шаг 1 (Приведение к 1НФ). На первом шаге задается одно или несколько отношений, отображающих понятия

предметной области. По модели предметной области выписываются обнаруженные функциональные зависимости. Все отношения автоматически находятся в 1НФ.

Шаг 2 (Приведение к 2НФ). Если в некоторых отношениях обнаружена зависимость атрибутов от части сложного ключа, то проводим декомпозицию этих отношений на несколько отношений следующим образом: те атрибуты, которые зависят от части сложного ключа выносятся в отдельное отношение вместе с этой частью ключа. В исходном отношении остаются все ключевые атрибуты.

Шаг 3 (Приведение к 3НФ). Если в некоторых отношениях обнаружена зависимость некоторых неключевых атрибутов от других неключевых атрибутов, то надо проводить декомпозицию этих отношений следующим образом: те неключевые атрибуты, которые зависят от других неключевых атрибутов выносятся в отдельное отношение. В новом отношении ключом становится детерминант функциональной зависимости.

В большинстве случаев третьей нормальной формы вполне достаточно, чтобы разрабатывать вполне работоспособные базы данных. Однако рассмотрим еще одну нормальную форму более высокого порядка, а именно, нормальную форму Бойса-Кодда (НФБК).

При приведении отношений при помощи нормализации к отношениям в 3НФ неявно предполагалось, что все отношения содержат один потенциальный ключ. Это не всегда верно.

Функциональная зависимость атрибутов утверждает лишь то, что для каждого конкретного состояния базы данных по значению одного атрибута (детерминанта) можно однозначно определить значение другого атрибута (зависимой части). Но конкретные значения зависимой части могут быть различными в различных состояниях базы данных.

Если имеются отношения, содержащие несколько потенциальных ключей, то необходимо проверить, имеются ли функциональные зависимости, детерминанты которых не яв-

ляются потенциальными ключами. Если такие функциональные зависимости имеются, то необходимо провести дальнейшую декомпозицию отношений. Те атрибуты, которые зависят от детерминантов, не являющихся потенциальными ключами выносятся в отдельное отношение вместе с детерминантами.

Отношение находится в нормальной форме Бойса-Кодда (НФБК) тогда и только тогда, когда детерминанты всех функциональных зависимостей являются потенциальными ключами.

Итак, для того, что бы проверить соответствие предварительных отношений требованиям нормальной формы Бойса-Кодда необходимо выявить все функциональные зависимости и все потенциальные ключи в каждом из отношений. Результаты подобного анализа видны из табл. 3.6, где потенциальные ключи выделены жирным шрифтом. В приведенном примере выявлены потенциальные ключи в отношениях Работники (Должность) и Заказы (НомАвт).

Очевидно, что декомпозиция отношений, в которых были обнаружены потенциальные ключи, приведет к более высокой степени их нормализации (НФБК).

Отношение находится в четвертой нормальной форме (4НФ) тогда и только тогда, когда отношение находится в НФБК и не содержит нетривиальных многозначных зависимостей.

Выполнение условий 4НФ приводит к независимости полей. Для отношения находящегося в 4НФ, изменение значений любого поля (не входящего в первичный ключ) должно выполняться без воздействия на данные других полей.

После нормализации отношений будет получен список окончательных отношений для рассматриваемой организации. Он имеет следующий вид, как показано в табл. 7.4.

Таблица 7.4

Таблица окончательных отношений

Название отношения	Ключевые поля, поля для связи и неключевые атрибуты	Нормальная форма
Работники	(ТабНом, <u>Фамилия</u> , <u>Имя</u> , <u>ДатаРожд</u> , Должность)	ЗНФ
Работы	(<u>НомРаботы</u> , <u>ТабНом</u> , <u>Название</u> , <u>ДатаНач</u> , <u>ДатаО-конч</u>)	НФБК
Клиенты	(<u>НомКлиента</u> , <u>Телефон</u> , <u>Адрес</u>)	НФБК
Заказы	(<u>НомЗаказа</u> , <u>ТабНом</u> , <u>НомКлиента</u> , <u>Дата</u> , <u>Сумма</u> , НомАвт)	ЗНФ
Поставщики	(<u>НомПоставщика</u> , <u>Название</u> , <u>КонтТел</u> , <u>Адрес</u>)	НФБК
Продукция	(<u>Артикул</u> , <u>НазвПрод</u> , <u>Вес</u> , <u>Цвет</u>)	НФБК
Поставляют	(<u>Артикул</u> , <u>НомПоставщика</u> , <u>ДатаПост</u> , <u>СтоимПост</u>)	НФБК

Данная таблица - Таблица окончательных отношений - является обязательной для отчета по курсовому проектированию. Очевидно, что количество окончательных отношений будет равно числу таблиц, участвующих в схеме данных.

8. ТЕМЫ КУРСОВЫХ РАБОТ

- 1 База данных Access «Банк»
- 2 База данных Access «Склад»
- 3 База данных Access «Автосалон»
- 4 База данных Access «Кинотеатр»
- 5 База данных Access «Салон красоты»
- 6 База данных Access «Страховая компания»
- 7 База данных Access «Гостиница»
- 8 База данных Access «Ломбард»
- 9 База данных Access «Реализация готовой про-
дукции»
- 10 База данных Access «Ведение заказов»
- 11 База данных Access «Бюро по трудоустройству»
- 12 База данных Access «Нотариальная контора»
- 13 База данных Access «Фирма по продаже запча-
стей»
- 14 База данных Access «Курсы повышения квали-
фикации»
- 15 База данных Access «Определение факультативов
для студентов»
- 16 База данных Access «Распределение учебной на-
грузки»
- 17 База данных Access «Распределение дополни-
тельных обязанностей»
- 18 База данных Access «Техническое обслуживание
станков»
- 19 База данных Access «Туристическая фирма»
- 20 База данных Access «Грузовые перевозки»
- 21 База данных Access «Учет телефонных перего-
воров»
- 22 База данных Access «Учет внутриофисных рас-
ходов»
- 23 База данных Access «Библиотека»
- 24 База данных Access «Прокат автомобилей»

- 25 База данных Access «Выдача кредитов»
- 26 База данных Access «Инвестирование свободных средств»
- 27 База данных Access «Занятость актеров театра»
- 28 База данных Access «Платная поликлиника»
- 29 База данных Access «Сдача в аренду торговых площадей»
- 30 База данных Access «Отдел кадров»
- 31 База данных Access «Магазин автомобильных запчастей»
- 32 База данных Access «Спортивный клуб»
- 33 База данных Access «Учет телекомпанией стоимости прошедшей в эфире рекламы»
- 34 База данных Access «Интернет-магазин»
- 35 База данных Access «Ювелирная мастерская»
- 36 База данных Access «Парикмахерская»
- 37 База данных Access «Химчистка»
- 38 База данных Access Поставка товаров
- 39 База данных Access Торговля
- 40 База данных Access Оплата коммунальных услуг
- 41 База данных Access Абонемент библиотеки
- 42 База данных Access Коммерческая фирма
- 43 База данных Access Издательский центр
- 44 База данных Access Зеленстрой
- 45 База данных Access Отдел кадров предприятия
- 46 База данных Access Розничная торговля
- 47 База данных Access Банковские вклады
- 48 База данных Access Аудит
- 49 База данных Access Фабрика
- 50 База данных Access Подписные издания
- 51 База данных Access Спецодежда
- 52 База данных Access ГАИ
- 53 База данных Access Интернет продажи
- 54 База данных Access Ремонт бытовой техники
- 55 База данных Access Мебель

ЗАКЛЮЧЕНИЕ

На этом заканчивается изучение СУБД MS Access в рамках дисциплины «Информационные системы в экономике». Оно охватывала все основные инструменты ввода, обновления и вывода данных. Если Вы внимательно читали данное руководство и выполняли задания, то теперь сможете достаточно свободно работать с реальными базами данных.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙ-
СКОЙ ФЕДЕРАЦИИ
ФГБОУ ВО «Воронежский государственный технический
университет»

Инженерно-экономический факультет

Кафедра экономики и управления на предприятии
машиностроения

КУРСОВАЯ РАБОТА

по дисциплине «Информационные системы в экономике»

Тема: « _____ »

Выполнил (а): ст-т (ка) гр. _____
индекс группы подпись Фамилия И.О.

Принял (а): _____
подпись Фамилия И.О.

Оценка: _____

Дата : _____

Воронеж 20__

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГБОУ ВО «Воронежский государственный технический университет»

Инженерно-экономический факультет

Кафедра экономики и управления на предприятии машиностроения

Студент _____
(индекс группы)

(фамилия, имя, отчество)

Задание на курсовую работу

1. Тема _____

2. Содержание _____

3. График выполнения

Раздел	Процент %	Срок	Процент выполнения %	Подпись руководителя

Руководитель _____

Допустить студента _____ к защите курсовой работы

Назначить защиту на _____

Оценка руководителя _____

ПРИЛОЖЕНИЕ 3

Примерное содержание курсовой работы по дисциплине «Информационные системы в экономике»

Введение

1. Постановка задачи
2. Проектирование базы данных
3. Создание базы данных в СУБД MS Access
 - 3.1. Создание таблиц базы данных и схемы данных
 - 3.2. Создание запросов
 - 3.3. Создание форм
 - 3.4. Создание отчетов

Заключение

Список использованных источников

Приложение 1

Приложение 2

Приложение 3

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Александров, Д.В. Инструментальные средства информационного менеджмента. CASE-технологии и распределенные информационные системы [Текст]: учеб. пособие / Д.В. Александров. - М.: ФиС, 2011. - 224 с.
2. Алиев, В.С. Информационные технологии и системы финансового менеджмента [Текст]: учеб. пособие / В.С. Алиев. - М.: Форум, ИНФРА-М, 2011. - 320 с.
3. Амириди, Ю.В. Информационные системы в экономике. Управление эффективностью банковского бизнеса [Текст]: учеб. пособие / Ю.В. Амириди, Е.Р. Кочанова, О.А. Морозова. - М.: КноРус, 2011. - 174 с.
4. Балдин, К.В. Информационные системы в экономике [Текст]: учебник / К.В. Балдин, В.Б. Уткин. - М.: Дашков и К, 2015. - 395 с.
5. Балдин, К.В. Информационные системы в экономике [Текст]: учеб. пособие (ГРИФ) / К.В. Балдин. - М.: ИНФРА-М, 2012. - 218 с.
6. Блиновская, Я.Ю. Введение в геоинформационные системы [Текст]: учеб. пособие / Я.Ю. Блиновская, Д.С. Задоя. - М.: Форум, НИЦ ИНФРА-М, 2013. - 112 с.
7. Бодров, О.А. Предметно-ориентированные экономические информационные системы [Текст]: учебник для вузов. / О.А. Бодров, Р.Е. Медведев. - М.: ГЛТ, 2013. - 244 с.
8. Бородакий, Ю.В. Информационные технологии. Методы, процессы, системы [Текст]: учеб. пособие / Ю.В. Бородакий, Ю.Г. Лободинский. - М.: ГЛТ, 2004. - 456 с.
9. Брусакова, И.А. Информационные системы и технологии в экономике [Текст] / И.А. Брусакова, В.Д. Чертовски. - М.: Финансы и статистика, 2007. - 352 с.
10. Буреш, О.В. Интеллектуальные информационные системы управления социально-экономическими объектами [Текст] / О.В. Буреш, М.А. Жук. - М.: Красанд, 2012. - 192 с.

11. Варфоломеева, А.О. Информационные системы предприятия [Текст]: учеб. пособие / А.О. Варфоломеева, А.В. Коряковский, В.П. Романов. - М.: НИЦ ИНФРА-М, 2013.- 283 с.
12. Васильков, А.В. Информационные системы и их безопасность [Текст]: учеб. пособие / А.В. Васильков, А.А. Васильков, И.А. Васильков. - М.: Форум, 2013. - 528 с.
13. Вдовин, В.М. Предметно-ориентированные экономические информационные системы [Текст]: учеб. пособие / В.М. Вдовин, Л.Е. Суркова и др. - М.: Дашков и К, 2016. - 388 с.
14. Гвоздева, В.А. Информатика, автоматизированные информационные технологии и системы [Текст]: учебник / В.А. Гвоздева. - М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2013. - 544 с.
15. Гвоздева, В.А. Информатика, автоматизированные информационные технологии и системы [Текст]: учебник (ГРИФ) / В.А. Гвоздева. - М.: Форум, 2011. - 544 с.
16. Гобарева, Я.Л. Банковские информационные системы и технологии. Ч. 1. Технология банковского учета [Текст] / Я.Л. Гобарева, Е.Р. Кочанова. - М.: Финансы и статистика, 2005. - 384 с.
17. Горбенко, А.О. Информационные системы в экономике [Текст]: учеб. пособие / А.О. Горбенко. - М.: БИНОМ. Лаборатория знаний, 2014. - 292 с.
18. Гришин, А.В. Промышленные информационные системы и сети: практическое руководство [Текст] / А.В. Гришин. - М.: Радио и связь, 2010. - 176 с.
19. Данелян, Т.Я. Экономические информационные системы (ЭИС) предприятий и организаций [Текст]: монография / Т.Я. Данелян. - М.: ЮНИТИ, 2015. - 284 с.
20. Дворкович, В.П. Цифровые видеоинформационные системы (теория и практика) [Текст] / В.П. Дворкович, А.В. Дворкович. - М.: Техносфера, 2012. - 1008 с.
21. Емельянов, С.В. Информационные технологии и вычислительные системы: Интернет-технологии. Математическое моделирование. Системы управления. Компьютерная графика [Текст] / С.В. Емельянов. - М.: Ленанд, 2015. - 96 с.

22. Енджейчик, И. Предметно-ориентированные экономические информационные системы [Текст]: учебник / И. Енджейчик. - М.: Финансы и статистика, 2007. - 224 с.
23. Ермолин, Н.П. Информационные системы в экономике. Практикум [Текст]: учебник / Н.П. Ермолин. - М.: КноРус, 2012. - 256 с.
24. Заварина, Е.С. Инструментальные средства информационного менеджмента. CASE-технологии и распределенные информационные системы [Текст]: учеб. пособие / Е.С. Заварина. - М.: Финансы и статистика, 2011. - 224 с.
25. Золотова, Е.В. Основы кадастра: Территориальные информационные системы [Текст]: учеб. пособие / Е.В. Золотова. - М.: Академический проект, 2012. - 416 с.
26. Иванкин, Е.Ф. Информационные системы с апостериорной обработкой результатов наблюдений. / Е.Ф. Иванкин. - М.: ГЛТ, 2009. - 168 с.
27. Исаев, Г.Н. Информационные системы в экономике [Текст]: учебник для студентов вузов / Г.Н. Исаев. - М.: Омега-Л, 2013. - 462 с.
28. Калашян, А.Н. Информационные системы в экономике: В 2-х ч. Ч.2. Практика использования [Текст]: учеб. пособие / А.Н. Калашян. - М.: Финансы и статистика, 2006. - 240 с.
29. Карминский, А.М. Информационные системы в экономике: В 2-х ч. Ч.1. Методология создания [Текст]: учеб. пособие / А.М. Карминский, Б.В. Черников. - М.: Финансы и статистика, 2006. - 336 с.
30. Карминский, А.М. Информационные системы в экономике: В 2-х ч. Ч.2. Практика использования [Текст]: учеб. пособие / А.М. Карминский, Б.В. Черников. - М.: Финансы и статистика, 2006. - 240 с.
31. Косиненко, Н.С. Информационные системы и технологии в экономике [Текст]: учеб. пособие / Н.С. Косиненко, И.Г. Фризен. - М.: Дашков и К, 2015. - 304 с.
32. Кувшинов, М.С. Информационные системы в экономике. Управление эффективностью банковского бизнеса

[Текст]: учеб. пособие / М.С. Кувшинов. - М.: КноРус, 2013. - 176 с.

33. Лихтенштейн, В.Е. Информационные технологии в бизнесе. Практикум: применение системы Decision в микро- и макроэкономике [Текст]: учеб. пособие / В.Е. Лихтенштейн. - М.: ФиС, 2009. - 512 с.

34. Мезенцев, К.Н. Автоматизированные информационные системы [Текст]: учебник для студентов учреждений среднего профессионального образования / К.Н. Мезенцев. - М.: ИЦ Академия, 2013. - 176 с.

35. Миков, А.И. Информационные процессы и нормативные системы в IT: Математические модели. Проблемы проектирования. Новые подходы [Текст]: учеб. пособие / А.И. Миков. - М.: КД Либроком, 2013. - 256 с.

36. Нестеров, В.В. Информационные системы в экономике: В 2-х ч. Ч.1. Методология создания [Текст]: учеб. пособие / В.В. Нестеров. - М.: Финансы и статистика, 2006. - 336 с.

37. Норенков, И.П. Автоматизированные информационные системы [Текст]: учеб. пособие / И.П. Норенков. - М.: МГТУ им. Баумана, 2011. - 342 с.

38. Одинцов, Б.Е. Информационные системы управления эффективностью бизнеса [Текст]: учебник и практикум для бакалавриата и магистратуры / Б.Е. Одинцов. - Люберцы: Юрайт, 2016. - 206 с.

39. Олейник, П.П. Корпоративные информационные системы [Текст]: учебник для вузов. Стандарт третьего поколения / П.П. Олейник. - СПб.: Питер, 2012. - 176 с.

40. Патров, В.В. Компьютерный практикум для менеджеров: информационные технологии и системы [Текст]: учеб. пособие / В.В. Патров. - М.: Финансы и статистика, 2006. - 352 с.

41. Патрушина, С.М. Информационные системы в экономике [Текст]: учеб. пособие / С.М. Патрушина, Н.А. Аручи-ди.. - М.: Мини Тайп, 2012. - 144 с.

42. Перова, М.Б. Информационные системы и технологии в экономике [Текст]: учеб. пособие / М.Б. Перова. - М.: Финансы и статистика, 2007. - 352 с.
43. Пипко, В.А. Информационные сети и системы. [Текст]: справочная книга / В.А. Пипко и др. - М.: Финансы и статистика, 1996. - 368 с.
44. Пирогов, В. Информационные системы и базы данных: организация и проектирование [Текст]: учеб. пособие: / В. Пирогов. - СПб.: BHV, 2009. - 528 с.
45. Починок, А.П. Банковские информационные системы и технологии. Ч.1. Технология банковского учета [Текст]: учеб. пособие / А.П. Починок и др. - М.: Финансы и статистика, 2005. - 384 с.
46. Раклов, В.П. Географические информационные системы в тематической картографии [Текст]: учеб. пособие / В.П. Раклов. - М.: Академический проект, 2014. - 176 с.
47. Реутов, А.П. Автоматизированные информационные системы: методы построения и исследования [Текст]: учеб. пособие / А.П. Реутов, М.В. Черняков, С.Н. Замуруев. - М.: Радиотехника, 2010. - 328 с.
48. Рубичев, Н.А. Измерительные информационные системы [Текст]: учеб. пособие / Н.А. Рубичев. - М.: Дрофа, 2010. - 334 с.
49. Рыбакова, О.В. Информационные системы и технологии в экономике [Текст]: учеб. пособие / О.В. Рыбакова. - М.: Финансы и статистика, 2006. - 416 с.
50. Рыжко, А.Л. Информационные системы управления производственной компанией [Текст]: учебник для академического бакалавриата / А.Л. Рыжко, А.И. Рыбников, Н.А. Рыжко. - Люберцы: Юрайт, 2016. - 354 с.
51. Семакин, И.Г. Информационные системы и модели. Элективный курс [Текст]: учеб. пособие / И.Г. Семакин. - М.: Бином. Лаборатория знаний, 2005. - 303 с.
52. Сухова, Л.Ф. Информационные технологии в бизнесе. Практикум: применение системы Decision в микро- и макро-

экономике [Текст]: учеб. пособие / Л.Ф. Сухова и др. - М.: Финансы и статистика, 2008. - 512 с.

53. Сырецкий, Г.А. Информатика. Фундаментальный курс. Том II. Информационные технологии и системы [Текст]: учеб. пособие / Г.А. Сырецкий. - СПб.: ВHV, 2012. - 848 с.

54. Уткин, В.Б. Информационные системы в экономике: [Текст]: учебник для студентов высших учебных заведений / В.Б. Уткин, К.В. Балдин. - М.: ИЦ Академия, 2012. - 288 с.

55. Федорова, Г.Н. Информационные системы [Текст]: учебник для студ. учреждений сред. проф. образования / Г.Н. Федорова. - М.: ИЦ Академия, 2013. - 208 с.

56. Федотова, Е.Л. Информационные технологии и системы [Текст]: учеб. пособие / Е.Л. Федотова. - М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2013. - 352 с.

57. Фельдман, А.Б. Информационные технологии в бизнесе. Практикум: применение системы Decision в решении прикладных экономических задач [Текст]: учеб. пособие / А.Б. Фельдман. - М.: Финансы и статистика, 2009. - 560 с.

58. Чадаев, В.К. Информационные системы компаний связи. Создание и внедрение [Текст]: учеб. пособие / В.К. Чадаев. - М.: Эко-Трендз, 2004. - 256 с.

59. Чандра, А.М. Дистанционное зондирование и географические информационные системы [Текст]: учеб. пособие / А.М. Чандра. - М.: Техносфера, 2008. - 312 с.

60. Ясенев, В.Н. Информационные системы и технологии в экономике [Текст]: учебное пособие для студентов вузов / В.Н. Ясенев. - М.: ЮНИТИ-ДАНА, 2012. - 560 с.

61. Титоренко, Г.А. Информационные системы в экономике [Текст]: учебник; под ред. Г.А. Титоренко. - М.: ЮНИТИ, 2009. - 463 с.

62. Тельнова, Ю.Ф. Информационные системы и технологии [Текст]: учеб. пособие; под ред. Ю.Ф. Тельнова. - М.: ЮНИТИ, 2012. - 303 с.

ОГЛАВЛЕНИЕ

Введение	3
1. БАЗЫ ДАННЫХ	6
1.1. Понятия и определения	6
1.2. Компоненты системы баз данных	7
1.3. Функции СУБД	10
1.4. Классификация СУБД	12
1.5. Краткие выводы	15
2. СУБД MS ACCESS	16
2.1. Общие сведения	16
2.2. Нормализация отношений	17
2.3. Проектирование БД	18
2.4. Физическая структура данных	21
2.5. Краткие выводы	23
3. ЦЕЛЬ И ЗАДАЧИ ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ	24
4. ОСНОВНЫЕ ТРЕБОВАНИЯ, ПРЕДЪЯВЛЯЕМЫЕ К КУРСОВОЙ РАБОТЕ	24
5. ЗАЩИТА КУРСОВОЙ РАБОТЫ	28
6. ПОРЯДОК РАЗРАБОТКИ РАЗДЕЛОВ КУРСОВОЙ РАБОТЫ ПОСТАНОВКА ЗАДАЧИ	28
6.1. Цель разработки базы данных	29
6.2. Требования к БД	30
6.3. Требования к надежности	32
6.4. Создание БД в MS Access	34
6.5. Оформление отчета	35
7. ОСНОВНЫЕ ПРАВИЛА ПРОЕКТИРОВАНИЯ БД	37
7.1. Правила построения ER-диаграмм	37
7.2. Правила получения предварительных отношений	48
7.3. Правила нормализаций отношений	62
8. ТЕМЫ КУРСОВЫХ РАБОТ	67
Заключение	69
Приложения	70
Библиографический список	73

Учебное издание

Лубянская Элина Борисовна
Лукаш Елена Николаевна

ИНФОРМАЦИОННЫЕ СИСТЕМЫ
В ЭКОНОМИКЕ: КУРСОВОЕ ПРОЕКТИРОВАНИЕ

В авторской редакции

Подписано в печать 22.11.2017.
Формат 60x84/16. Бумага для множительных аппаратов.
Усл. печ. л. 5,0. Уч.-изд. л. 3,8. Тираж 350 экз.
Зак. №

ФГБОУ ВО «Воронежский государственный технический
университет»
394026 Воронеж, Московский просп., 14

Участок оперативной полиграфии издательства ВГТУ
394026 Воронеж, Московский просп., 14