

Лекция 13. Проектирование БД в MS Access. Структура таблиц. Типы данных. Схема данных. Модификация структуры данных.

СУБД Access оперирует реляционными базами данных.

На этапе проектирования базы данных должна быть **определена логическая структура** базы данных для выбранной предметной области. **Проект логической структуры** БД устанавливает **состав реляционных таблиц, их структуру и логические связи между таблицами**. При формировании структуры каждой **таблицы** определяется **совокупность атрибутов (полей, столбцов)**, для каждого из которых даются описание типа, размера данных и других свойств. Кроме того, должен быть указан **уникальный ключ таблицы**, который может состоять из одного или нескольких полей. При проектировании базы данных, отвечающей **требованиям нормализации**, между таблицами определяются **логические связи типа 1 : М**. Такие связи позволят осуществлять в Access автоматическое поддержание **связной целостности и непротиворечивости** данных в базе.

Этапы проектирования и создания базы данных

Для проектирования базы данных необходимо располагать **описанием выбранной предметной области**, которое должно охватывать **реальные объекты и процессы, определять все необходимые источники информации** для обеспечения предполагаемых запросов пользователя и решаемых в приложении задач.

Определение **состава и структуры данных**, которые должны быть загружены в базу данных, осуществляется на основе анализа предметной области. Структура данных предметной области может отображаться информационно-логической моделью (ИЛМ). Если при построении такой модели обеспечены требования нормализации данных и она, соответственно, представлена в каноническом виде, далее легко определяется **проект логической структуры нормализованной базы данных**. На основе канонической модели можно создать реляционную базу без дублирования данных.

При разработке **модели данных предметной области** могут использоваться **два подхода**.

В первом подходе (аналитическом или процессном) сначала формулируются основные задачи, для решения которых строится база, выявляются информационные потребности задач приложения пользователя, и, соответственно, определяются состав и структура информационных объектов модели, а также связи между ними.

При втором подходе (интуитивном) сразу устанавливаются типовые объекты предметной области и их взаимосвязи.

Наиболее рационально **сочетание обоих подходов**. Это связано с тем, что на начальном этапе, как правило, нет исчерпывающих сведений обо всех задачах. Использование такой технологии тем более оправдано, что гибкие средства создания реляционной базы данных в Access позволяют на любом этапе разработки внести изменения в базу данных и модифицировать ее структуру без ущерба для введенных ранее данных.

При определении **проекта логической структуры** реляционной базы данных **каждый информационный объект канонической модели предметной области адекватно отображается реляционной таблицей, а связям между двумя информационными объектами соответствуют логические связи между парой соответствующих таблиц.**

Такие **связи устанавливаются по уникальному ключу одной** из этих таблиц, которая является **главной** в связи. Во второй таблице, которая является подчиненной, поле связи может быть **либо частью ее уникального ключа, либо неключевым.**

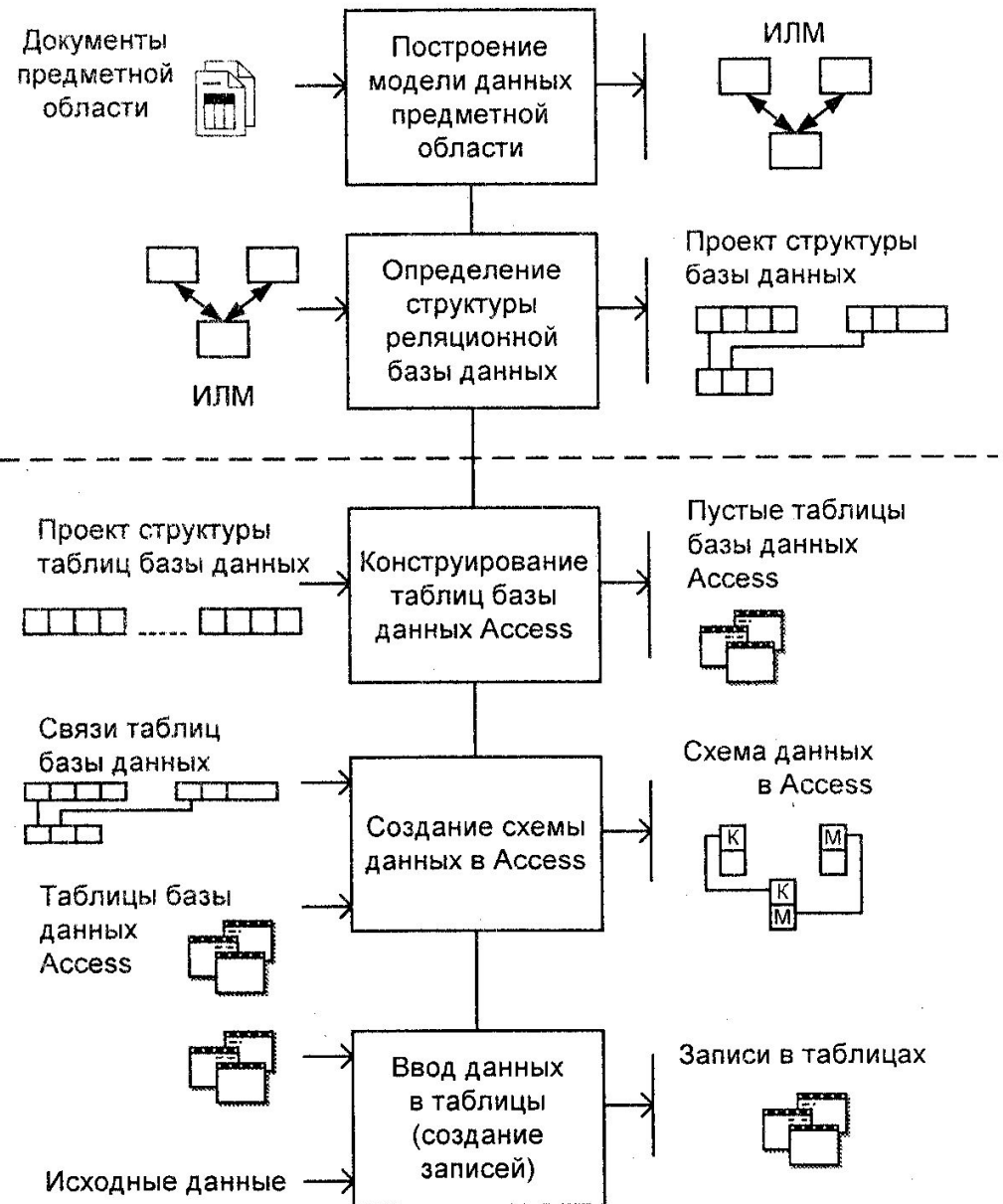
В процессе создания базы данных на компьютере **сначала осуществляется конструирование ее таблиц средствами Access**. Далее создается **схема данных**, в которой устанавливаются **логические связи таблиц**. В **схеме данных базы могут быть заданы параметры поддержания связной целостности данных**, если модель данных была разработана в соответствии с требованиями нормализации.

Информационно-логическая модель (ИЛМ) отображает данные предметной области в виде **совокупности информационных объектов (ИО) и связей между ними**. Эта модель представляет данные, подлежащие хранению в базе данных. Каждый информационный объект в модели данных должен иметь уникальное имя.

Разработка БД в Access

ПРОЕКТ

СОЗДАНИЕ



Свойства таблиц

Для определения поля таблицы задаются **Имя поля** (Field Name), **Тип данных** (Data Type), **Описание** (Description) — краткий комментарий, а также свойства поля в разделе **Свойства поля** (Field Properties), представленные на двух вкладках: **Общие** (General) и **Подстановка** (Lookup).

К **общим** относятся такие свойства поля, как максимальный размер, формат, подпись, которая выводится в заголовке столбца таблицы, значение по умолчанию и др. На вкладке **Подстановка** (Lookup) выбирается **Тип элемента управления (Display Control)**: поле, список фиксированных значений или поле со списком.

Имена полей и типы данных

Имя поля (Field Name). Каждое поле в таблице должно иметь уникальное имя, удовлетворяющее соглашениям об именах объектов в Access. Оно является комбинацией из букв, цифр, пробелов и специальных символов, за исключением точки (.), восклицательного знака (!), надстрочного знака (~) и квадратных скобок ([]). Имя не может начинаться с пробела и содержать управляющие символы с кодами ASCII от 0 до 31. Максимальная длина имени 64 символа.

Тип данных (Data Type). Тип данных определяется значениями, которые предполагается хранить в поле, и операциями, которые будут выполняться с этими значениями. В Access допускается использование **десяти типов данных и мастера подстановок**. Список возможных типов данных вызывается нажатием кнопки списка при выборе типа данных каждого поля. Рассмотрим вкратце назначение и допустимые размеры всех типов данных, которые могут назначаться полям таблицы в Access.

Текстовый (Text) — используется для хранения текста или комбинаций алфавитно-цифровых знаков, не применяемых в расчетах (например, код товара). Максимальная длина поля 255 знаков.

Поле MEMO (Memo) — используется для хранения обычного текста или комбинаций алфавитно-цифровых знаков длиной более 255 знаков. Поля с этим типом данных в базах данных формата Access 2007 поддерживают также форматирование текста. Это единственный в Access тип данных, обеспечивающий встроенную поддержку отображения и хранения форматированного текста. Максимальный размер поля 1 Гбайт знаков или 2 Гбайт памяти (2 байта на знак) при программном заполнении полей, и 65 535 знаков при вводе данных вручную в поле и в любой элемент управления, связанный с этим полем.

Числовой (Number) — служит для хранения числовых значений (целых или дробных), предназначенных для вычислений, исключением являются денежные значения, для которых используется тип данных **Денежный**. Размер поля 1, 2, 4 и 8 байтов, или 16 байтов (если используется для кода репликации) зависит от типа чисел, вводимых в поле.

Дата/время (Date/Time) — используется для хранения значений даты и времени в виде 8-байтовых чисел двойной точности с плавающей запятой. Целая часть значения, расположенная слева от десятичной запятой, представляет собой дату. Дробная часть, расположенная справа от десятичной запятой, — это время. Хранение значений даты и времени в числовом формате позволяет выполнять различные вычисления с этими данными.

Денежный (Currency) — используется для хранения денежных значений в виде 8-байтовых чисел с точностью до четырех знаков после запятой. Этот тип данных применяется для хранения финансовых данных и, в тех случаях, когда значения не должны округляться.

Счетчик (AutoNumber) — используется для уникальных числовых 4-байтовых значений, которые автоматически вводит Access при добавлении записи. Вводимые числа могут последовательно увеличиваться на указанное приращение или выбираться случайно. Обычно используются в первичных ключах.

Логический (Yes/No)— применяется для хранения логических значений, которые могут содержать одно из двух значений: Да/Нет, Истина/Ложь или Вкл/Выкл. (8 битов = 1 байт). Используется 1 для значений Да и 0 для значений Нет. Размер равен 1 биту.

Поле объекта OLE (OLE Object) — используется для хранения изображений, документов, диаграмм и других объектов из приложений Ms Office и других программ Windows в виде растровых изображений, которые затем отображаются в элементах управления форм или отчетов, связанных с этим полем таблицы. Чтобы в Access просмотреть эти изображения, необходимо, чтобы на компьютере, использующем базу данных, был зарегистрирован OLE-сервер (программа, поддерживающая этот тип файлов). Если для данного типа файлов OLE-сервер не зарегистрирован, отображается значок поврежденного изображения.

В accdb-файлах вместо типа данных **Поле объекта OLE** используется тип **Гиперссылка** (Hyperlink).

Гиперссылка (Hyperlink) — применяется для хранения ссылок на узлы (URL-адреса), на узлы или файлы интрасети или локальной сети (UNC-адреса — стандартного формата записи пути), а также на узлы или файлы локального компьютера. Кроме того, можно использовать ссылку на объекты Access, хранящиеся в базе данных. Может хранить до 1 Гбайт данных.

Вложение (Attachment) — используется для вложения в поле записи изображений, файлов электронных таблиц, документов, диаграмм и других файлов поддерживаемых типов точно так же, как в сообщения **электронной почты**. Вложенные файлы можно просматривать и редактировать в соответствии с заданными для поля параметрами. Эти поля не имеют ограничений, связанных с отсутствием зарегистрированных OLE -серверов. Более рационально используют место для хранения, чем поля с типом данных **Поле объекта OLE** (OLE Object), поскольку не создают растровые изображения исходного файла. Максимальная длина поля для сжатых вложений — 2 Гбайт, для несжатых — примерно 700 Кбайт в зависимости от степени возможного сжатия вложения.

Мастер подстановок (Lookup Wizard) — появляется в списке типов данных только в режиме конструктора. Фактически типом данных не является, а вызывает мастер подстановок, с помощью которого можно создать поле, позволяющее выбрать значения из списка, построенного на основе значений поля другой таблицы, запроса или фиксированного набора значений. Такое поле отображается как поле со списком. Если список построен на основе поля таблицы или запроса, тип данных и размер создаваемого поля определяется типом данных и размером привязанного столбца, если на основе набора значений — размером текстового поля, содержащего значение.

Замечание.

Код репликации (Replication ID). Глобальный уникальный идентификатор (Globally unique identifier, GUID), занимает 16 байтов. Эти длинные генерируемые случайным образом значения обеспечивают малую вероятность их совпадения. Поля такого типа используются Access для создания системных уникальных идентификаторов реплик, наборов реплик, таблиц, записей и других объектов при репликации баз данных. Могут быть использованы в приложениях пользователя, например, для идентификации товаров.

Общие свойства поля

Основные свойства задаются для каждого поля на вкладке **Общие (General)** и зависят от выбранного типа данных. Для отображения свойств поля необходимо установить курсор на строке соответствующего поля. Приведем свойства полей, наиболее важные на первом этапе изучения баз данных.

-Размер поля (Field Size) позволяет для текстового и числового поля уточнить тип данных или размер, задает максимальный размер данных, сохраняемых в поле. Для поля с типом данных **Текстовый (Text)** задается размер от 1 до 255 знаков. Изменения в данных, которые происходят вследствие изменения свойства **Размер поля (Field Size)**, нельзя отменить после выполнения сохранения инструкторе таблиц.

-Формат поля (Format) является форматом отображения выбранного типа данных при выводе их на экран или печать в режиме таблицы, в форме или отчете. В Access определены встроенные стандартные форматы отображения для полей с такими типами данных, как **Числовой** (Number), **Дата/время** (Date/Time), **Логический** (Yes/No) и **Денежный** (Currency). Ряд этих форматов совпадает с настройкой региональных форматов, определяемых в окне **Язык и региональные стандарты** в Панели управления Windows. Пользователь может создать собственный формат для всех типов данных, кроме **Поле объекта OLE** (OLE Object), с помощью символов форматирования. Для указания конкретного формата отображения необходимо выбрать в раскрывающемся списке одно из значений свойства **Формат поля** (Format).

-Число десятичных знаков (DecimalPlaces) задает для числового и денежного типов данных количество знаков после запятой. Можно задать число от 0 до 15. По умолчанию (значение **Авто** (Auto)) это число определяется установкой в свойстве **Формат поля** (Format). Следует иметь в виду, что установка этого свойства не действует, если свойство **Формат поля** (Format) не установлено или выбрано значение **Основной** (General Number). Свойство **Число десятичных знаков** (DecimalPlaces) влияет только на количество десятичных знаков, отображаемых на экране, и не влияет на число сохраняемых десятичных знаков. Для изменения числа сохраняемых знаков нужно изменить свойство **Размер поля** (Field Size).

-Подпись (Caption) поля задает текст, который выводится в таблицах, формах, отчетах.

-Значение по умолчанию (DefaultValue) определяет текст или выражение, значение которого автоматически вводится в поле при создании новой записи. Например, если задана функция =Now (), то в поле введется текущая дата и время. При добавлении записи в таблицу можно оставить значение, введенное по умолчанию, или ввести другое. Свойство **Значение по умолчанию** (DefaultValue) используется только при создании новой записи. Максимальная длина значения свойства составляет 255 знаков. Свойство не определено для полей с типом данных **Счетчик** (AutoNumber) или **Поле объекта OLE** (OLE Object).

-Условие на значение (ValidationRule) позволяет осуществлять контроль ввода, задает ограничения на вводимые значения, при нарушении условий запрещает ввод и выводит текст, заданный свойством **Сообщение об ошибке** (ValidationText).

Схема данных.

Реляционная база данных, созданная в соответствии с проектом канонической модели данных предметной области, состоит из **нормализованных таблиц, связанных отношениями 1-М.**

В такой базе данных обеспечивается **отсутствие дублирования описательных данных**, их однократный ввод, **поддержание целостности** данных средствами системы.

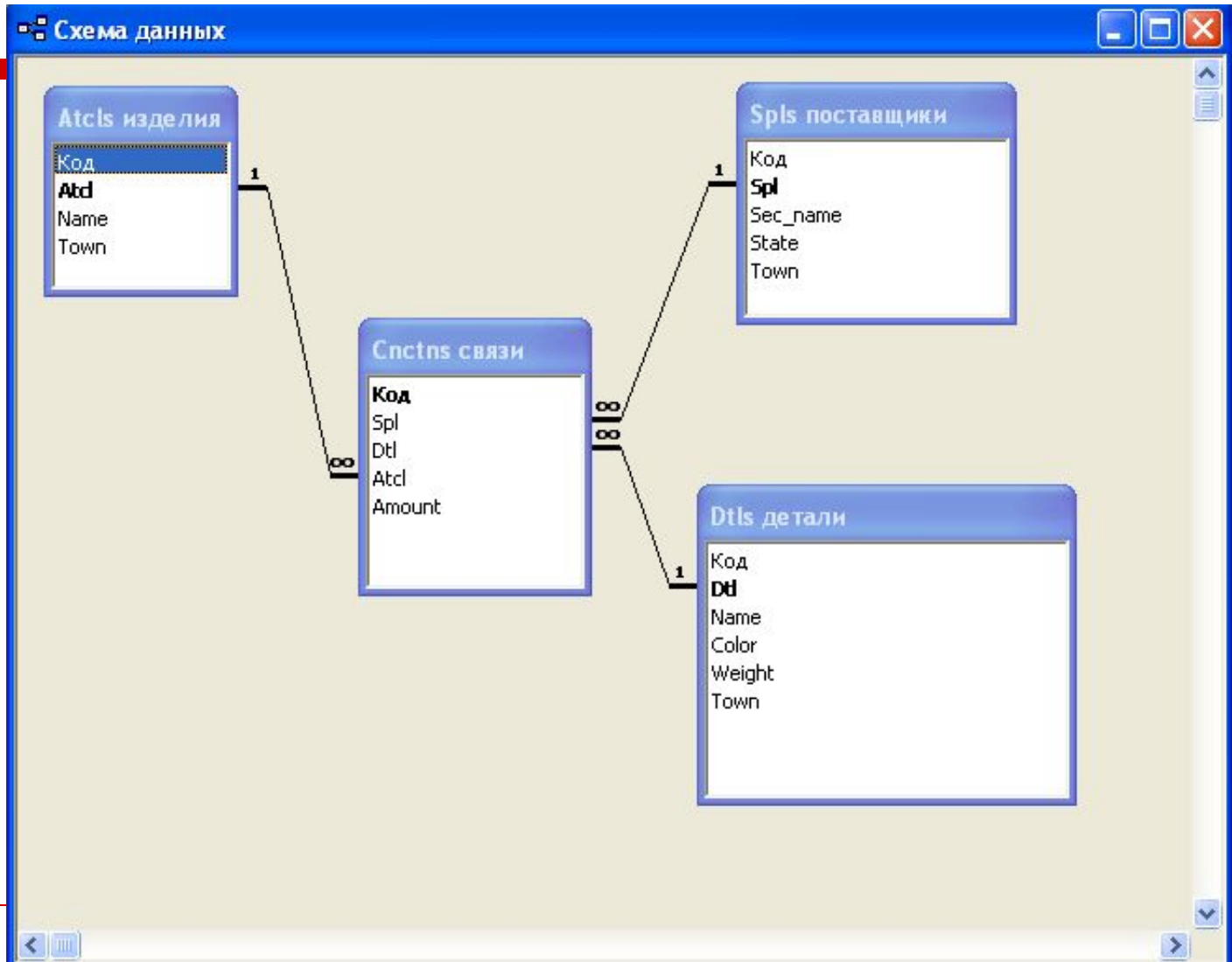
Связи между таблицами позволяют выполнить объединение данных различных таблиц, необходимое для решения большинства задач ввода, просмотра и корректировки данных, получения информации по запросам и вывода отчетов.

Связи между таблицами устанавливаются в соответствии с **проектом логической структуры** базы данных и запоминаются в схеме данных Access.

Схема данных является не только средством графического отображения логической структуры базы данных, она **активно используется системой в процессе обработки данных**. Система при необходимости обработки данных взаимосвязанных таблиц **автоматически использует связи, определенные в схеме данных**.

Таким образом, разработчику нет необходимости специально сообщать системе о наличии той или иной связи. Однажды указанные в схеме данных связи используются системой автоматически. Создание схемы данных позволяет упростить конструирование многотабличных форм, запросов, отчетов, а также обеспечить поддержание целостности взаимосвязанных данных при вводе и корректировке данных в таблицах.

Схема данных в Access



Создание связей между таблицами схемы данных

При создании связей в схеме данных используется **проект логической структуры реляционной базы данных**, в котором показаны все однозначные связи таблиц.

Реализуются связи с помощью **добавления в связанные таблицы общих полей, называемых ключом связи**.

При 1-М отношениях между таблицами ключом связи является ключ главной таблицы (простой или составной). В подчиненной таблице он может быть частью уникального ключа или вовсе не входить в состав ключа таблицы. 1-М связи являются основными в реляционных базах данных. 1-М связи используются, как правило, при необходимости распределять большое количество полей, определяемых одним и тем же ключом, по разным таблицам, имеющим разный регламент обслуживания.

Обеспечение целостности данных

Если связываемые в схеме данных таблицы находятся в отношении **1 : 1** и **1 : М**, для связи можно задать **параметр обеспечения связной Целостности данных**.

Обеспечение связной целостности данных означает, что Access при корректировке базы данных **реализует для связанных таблиц контроль соблюдения следующих условий**:

- в подчиненную таблицу **не может быть** добавлена запись с несуществующим в записях главной таблицы значением ключа связи;
 - в главной таблице **нельзя удалить** запись, если не удалены связанные с ней записи в подчиненной таблице;
 - **изменение значений ключа связи** в записи главной таблицы **невозможно**, если подчиненная таблица имеет связанные с ней записи, в которых соответственно не меняется ключ связи.
-

При **попытке пользователя нарушить** эти условия в операциях добавления и удаления записей или обновления ключевых данных в связанных таблицах **Access не допускает выполнения операции** и выводит соответствующее сообщение. Access не позволяет установить параметр целостности для связи таблиц, если ранее введенные в таблицы данные не отвечают требованиям целостности.

Установление между двумя таблицами связи и задание для нее параметров целостности данных возможно только при следующих условиях:

- связываемые поля имеют одинаковый тип данных, причем имена полей могут быть различными;
 - обе таблицы сохраняются в одной базе данных Access;
 - главная таблица связывается с подчиненной по первичному простому или составному ключу или уникальному индексу главной таблицы.
-

Каскадное обновление и удаление связанных записей

Если для выбранной связи обеспечивается поддержание целостности, можно задать режим каскадного удаления связанных записей и каскадного обновления связанных полей. Такие параметры делают возможным в главной таблице, соответственно, удаление записей и изменение значения в ключевом поле, т. к. при этих параметрах система автоматически выполнит необходимые изменения в подчиненных таблицах, обеспечив сохранение свойств целостности базы данных.

В режиме **каскадного удаления связанных записей** при удалении записи из главной таблицы будут автоматически удаляться все связанные записи в подчиненных таблицах. При удалении записи из главной таблицы выполняется каскадное удаление подчиненных записей на всех уровнях, если этот режим задан на каждом уровне.

В режиме **каскадного обновления связанных полей** при изменении значения ключевого поля в записи главной таблицы Access автоматически изменит значения в соответствующем поле в подчиненных записях.

Модификация структуры базы данных

Понятие "модификация структуры базы данных" включает **изменение структуры отдельных таблиц, добавление и удаление таблиц, а также изменение схемы данных.**

Изменение структуры таблиц

Для изменения структуры таблиц основным является режим конструктор таблиц, хотя некоторые изменения можно произвести в режиме таблицы. Такие простые операции, как изменение имени, некоторых типов данных, добавление, удаление неключевых полей, добавление полей со списком, могут быть произведены в обоих режимах. Однако в режиме таблицы и эти операции выполняются с различными ограничениями. Например, можно в режиме таблицы изменить имя поля, дважды щелкнув в его заголовке и введя новое значение. Это значение становится именем поля, но при этом теряется значение подписи поля.

Изменение полей, которые не являются ключами или полями связи

Состав и последовательность, а также тип данных, свойства или имена таких полей можно изменять независимо от наличия связей таблицы с другими таблицами базы данных. Для заполненной таблицы при изменении типа данных, размера и других свойств, производится преобразование данных. Однако, если преобразования недопустимы, попытка изменения типа данных может привести к потере данных, поэтому система запрашивает у пользователя подтверждение его действий.

Изменение или удаление ключевого поля

Для загруженной и несвязанной таблицы при попытке изменить свойства или удалить ключевое поле система предупреждает о возможной потере данных удалении ключа. Если, например, удаляется поле в составном ключе, с других полей этого ключа будет снят признак ключа. При попытке назначить ключом другое поле, имеющее повторяющиеся значения в загруженной таблице, операция не будет завершена и будет выдано сообщение о невозможности.

Если надо изменить ключ таблицы, которая имеет связи с другими таблицами в схеме данных, необходимо предварительно разорвать связи.

Изменение схемы данных

При модификации схемы данных может производиться изменение состава ее таблиц — удаление, добавление таблиц, изменение их связей, способа объединения записей связанных таблиц.

Необходимость в изменении связей возникает, в частности, при изменении ключей в таблицах. Изменение ключа по составу, по типу и размеру его полей не может производиться до тех пор, пока не удалены связи таблицы и схеме данных.

При изменении типа данных для неключевых полей, задействованных в связях таблицы, также предварительно необходимо удалить эти связи в схеме данных.

Для внесения изменений в схему данных перед ее открытием надо закрыть все таблицы. Команда **Схема данных** на вкладке **Работа с базами данных** в группе **Показать или скрыть** откроет схему данных. При этом открывается вкладка ленты **Работа со связями | Конструктор**.
