

6. Связывание таблиц с помощью отношений

В реальных практических базах данных таблицы соединены друг с другом паутиной связей. Одна из основных целей любой БД – разбить информацию на четкие управляемые порции. В хорошо спроектированной БД процесс завершается созданием большого числа таблиц. Несмотря на то, что в каждой из них записано что-то свое, часто придется путешествовать из одной таблицы в другую, чтобы получить всю нужную информацию.

Избыточные данные в противоположность связанным данным создают очевидную проблему. Из-за желания хранить слишком много подробностей в одном месте в одной таблице объединяется информация, которую лучше всего хранить в двух отдельных таблицах. Для исправления этого проекта нужно создать две таблицы со связанными данными. В программу Access включено средство, пытающееся отследить дублирующиеся данные в таблице и помочь разделить поля на две связанные таблицы. Другими словами, взаимосвязь таблиц предоставляет больше гибкости, позволяя задавать больше вопросов, касающихся данных, и получать более полные ответы.

Связывание с помощью столбца Код

Трудно найти данные, гарантированно уникальные и неизменные. Вместо риска возникновения проблем надежнее положиться на поле с типом данных *Счетчик*, содержащее код, генерируемый программой Access.

Интересно, что тот же подход применяется при связывании таблиц. Для ссылки на запись в другой таблице не следует использовать любую порцию данных – вместо этого нужно применять уникальный код, указывающий на нужную запись.

Отношение типа "родитель - потомок"

Разработчики БД применяют обозначения "родитель" и "потомок" для идентификации двух таблиц в отношении и определения каждой из них.

Вот чем объясняется такая аналогия. В реальной жизни у родителя может быть любое количество детей. А у ребенка только один "комплект" родителей. Это же правило действует и в БД.

Важно понять, что нельзя поменять местами таблицу-родитель и таблицу-потомок без изменения отношения между ними.

Если возникают трудности при определении родительской таблицы, существует простое правило, направляющее в нужное русло. В дочерней таблице всегда содержится порция идентифицирующей информации из родительской таблицы.

Этот же тип связи называют отношением "один - ко - многим", где один представляет родителя, а многие – детей, поскольку единственная родительская запись в одной таблице может быть связана с несколькими дочерними записями в другой. Это самое распространенное отношение, но не единственное.

Определение отношения

Действия, необходимые для определения связи или отношения:

- Любое отношение связывает два поля, находящиеся в разных таблицах. Сначала нужно определить в родительской таблице поле, которое следует использовать. В хорошо спроектированной БД в родительской таблице применяется поле первичного ключа.
- Открыть дочернюю таблицу в *Конструкторе*. Самый быстрый способ – щелкнуть правой кнопкой мыши таблицу в *Области переходов* и выбрать строку *Конструктор*.
- Создать нужное поле в таблице-потомке, если его еще нет.

Каждая дочерняя запись создает ссылку, сохраняя порцию данных, указывающую на запись в таблице-родителе. Нужно вставить новое поле для хранения этой информации, как показано на рис. 6.1.

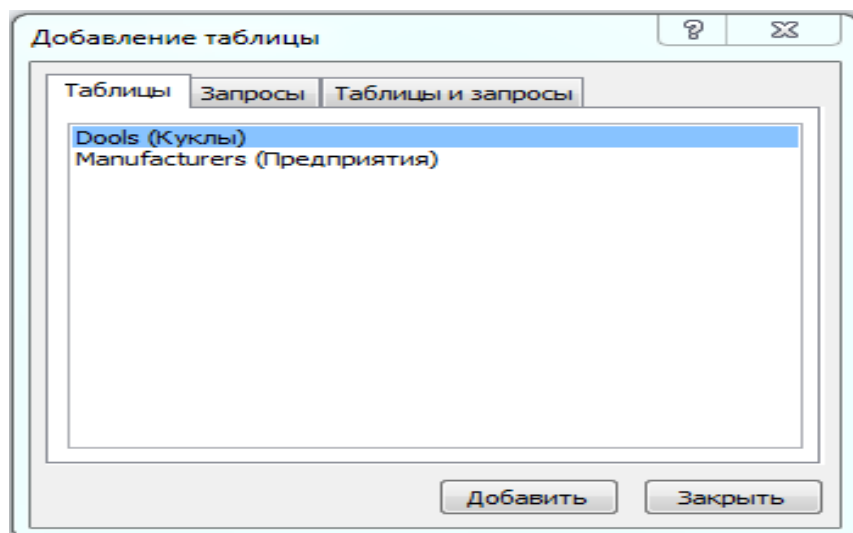


Рис. 6..1. Добавление таблиц в схему данных

У полей, которые связываются в родительской и дочерней таблицах, должны быть совместимые типы данных. Но существует небольшое затруднение. В поле родительской таблицы используется тип данных *Счетчик*, а поле дочерней таблицы должно быть полем *Числового* типа с размером поля *Длинное целое*. На диске типы *Счетчик* и *Длинное целое* в действительности хранят одинаковые данные. Но тип *Счетчик*, когда создается запись, сообщает программе Access о необходимости заполнения поля новым, автоматически генерируемым значением.

Сначала следует закрыть обе таблицы, программа Access предложит сохранить внесенные изменения, теперь таблицы готовы к установке связи, для этого следует выбрать *Работа с базами данных* → *Показать или скрыть* → *Схема данных*.

Программа Access откроет на ленте новую вкладку *Работа со связями*. Это специализированное окно, в котором можно определить связи между всеми таблицами БД. Прежде чем программа Access разрешит работать на вкладке *Работа со связями*, она выведет на экран диалоговое окно *Добавление таблицы*, запрашивая необходимые для работы таблицы.

На вкладке *Работа со связями* можно отобразить сколько угодно таблиц. Нужно следить за тем, чтобы не вставить одну и ту же таблицу дважды.

Вставить в рабочую область таблицу-родитель и таблицу-потомок.

Порядок указания таблиц не важен. Для выбора таблицы указать ее в списке и щелкнуть мышью кнопку *Добавить* или просто дважды щелкнуть кнопкой мыши имя таблицы.

Любая таблица на вкладке *Работа со связями* отображается в виде небольшого прямоугольника, в котором перечислены все поля таблицы. Если между таблицами уже определены связи, они отображаются как соединительные линии. Щелкнуть мышью кнопку *Закрыть*.

Теперь можно разместить таблицы на вкладке *Работа со связями* (рис. 6.2). На этой вкладке отображается схема данных – холст, на котором создаются устанавливаемые связи.

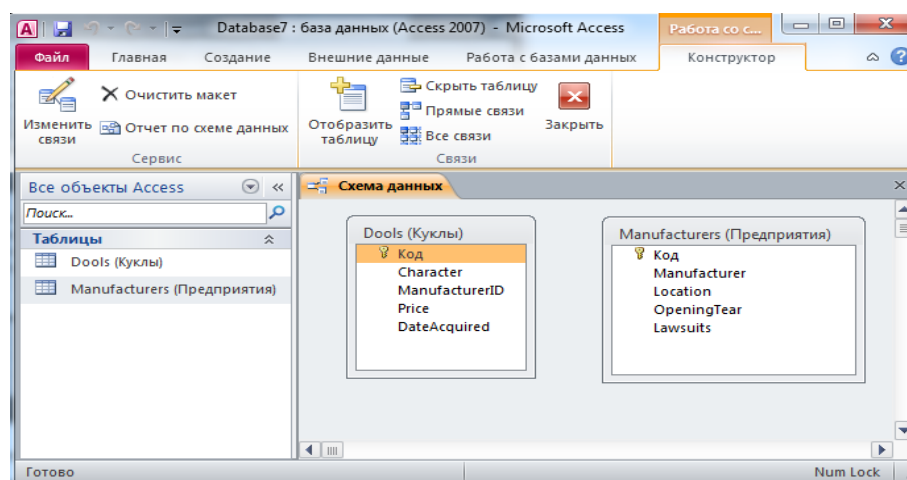


Рис.6.2. Расположение таблиц в схеме данных

Вставленные таблицы можно переместить с помощью мыши в любое место окна. Если в БД много связей, данная возможность позволит упорядочить таблицы таким образом, чтобы были четко видны все связи. Для удаления таблицы из схемы данных щелкнуть ее правой кнопкой мыши и выбрать команду *Скрыть таблицу*. Для вставки другой таблицы щелкнуть правой кнопкой мыши свободное пространство и выбрать команду *Добавить таблицу* для вывода на экран диалогового окна *Добавление таблицы*.

Для определения связи найти поле, которое используется в таблице-родителе. Перетащить это поле на поле в таблице-потомке, с которым оно связывается.

Можно перемещать и по-другому - из потомка в родителя. В любом случае программа Access создает одинаковую связь, или отношение.

Если при перемещении таблицы отпустить кнопку мыши, на экране появится диалоговое окно *Изменение связей* (рис. 6.3).

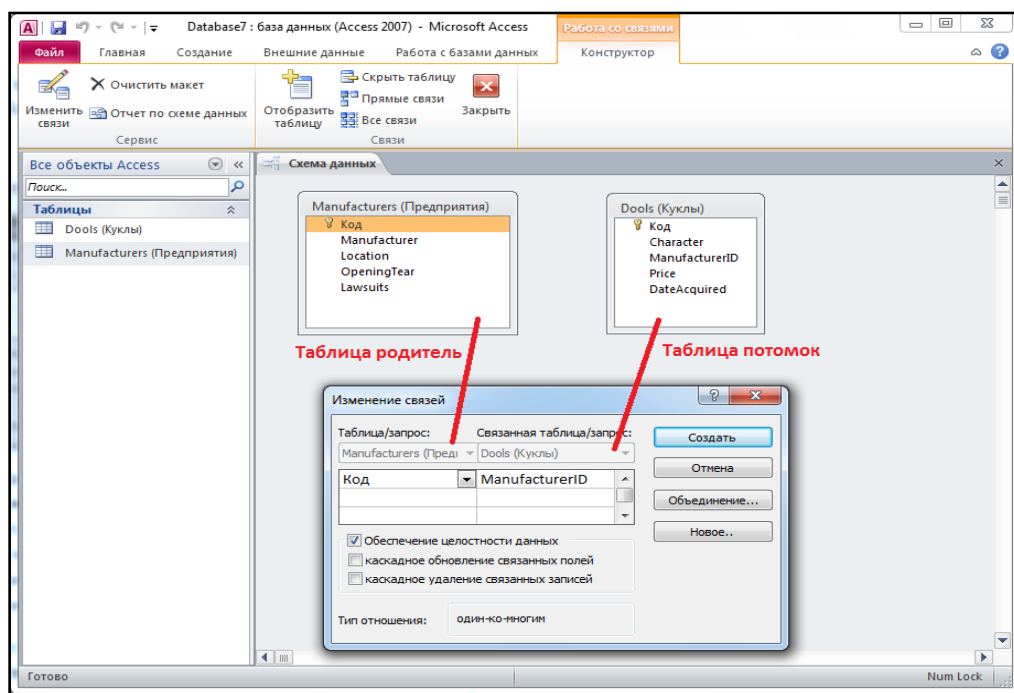


Рис.6.3. Изменение связей в схеме данных

Программа Access способна правильно определить таблицу-родитель, показанную в поле *Таблица/запрос*, и таблицу-потомок, показанную в поле *Связанная таблица/запрос*, когда соединяются два поля. Она идентифицирует поле в родительской таблице, поскольку у нее есть первичный ключ или уникальный индекс. Если в диалоговом окне *Изменение связей* что-то не так, прежде чем продолжить, можно поменять местами таблицы или заменить поля, используемые для создания связи.

Если необходимо предотвратить возможные ошибки, следует установить флажок *Обеспечение целостности данных*. Этот параметр включает улучшенную проверку ошибок, запрещающую внесение изменений, способных нарушить правила связи. Щелкнуть мышью кнопку *Создать*. Это действие создаст связь, соединяющую две таблицы. Связь появляется на схеме в виде линии (рис. 6.4).

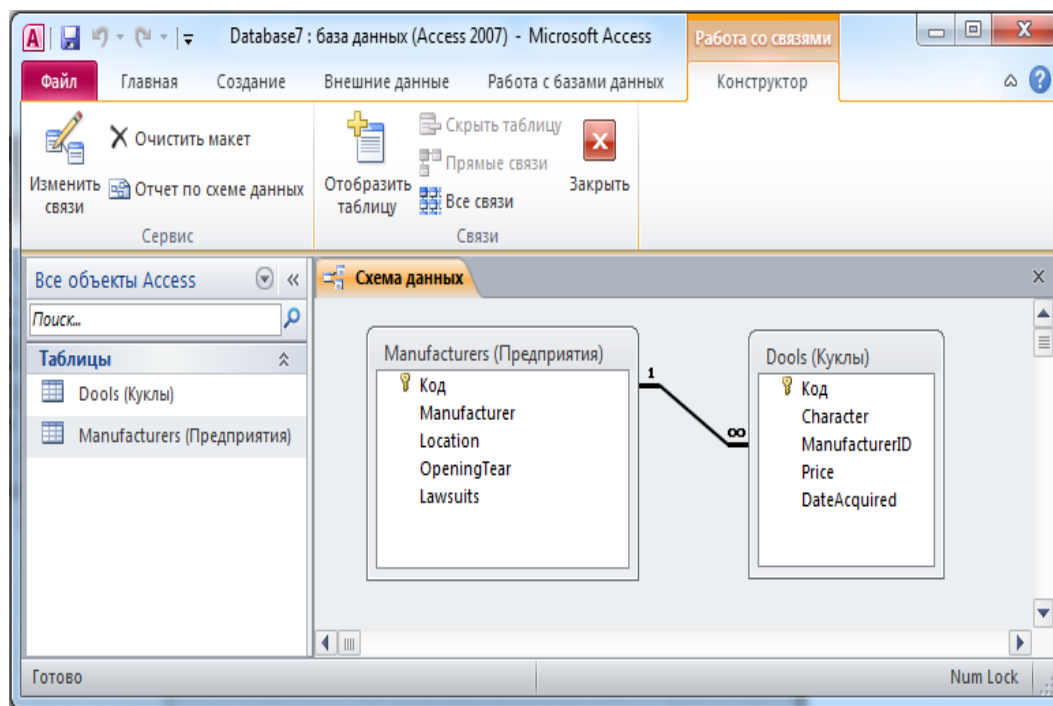


Рис.6.4. Создание связей

Если установить флажок *Обеспечение целостности данных*, программа Access проверит все имеющиеся в таблице данные на соблюдение правил связи. Если обнаружатся данные, их нарушающие, программа предупредит о проблеме и откажется продолжать. В этот момент лучшая стратегия – создать связь без обеспечения целостности, откорректировать неверные данные и позже отредактировать связь, установив флажок *Обеспечение целостности данных*.

Программа Access соединяет линией связанные поля на вкладке *Работа со связями*, маленькие значки 1 и бесконечность (∞) дают возможность в данном отношении "один-ко-многим" различить стороны "один" и "многие". Для редактирования связи дважды щелкнуть кнопкой мыши линию связи. Для того чтобы полностью удалить ее, щелкнуть дважды кнопкой мыши линию связи и выбрать команду *Удалить*

Чтобы закрыть вкладку *Работа со связями*, можно щелкнуть мышью кнопку в правом верхнем углу вкладки или выбрать *Работа со связями* | *Конструктор* → *Связи* → *Закрыть*.

Программа Access спросит, хотите ли сохранить макет вкладки *Работа со связями*. На самом деле речь идет о сохранении созданной схемы данных. Неважно, что выбрать, связь останется в БД, и можно использовать ее тем же способом. Единственное

различие заключается в возможности быстрого просмотра или редактирования связи на вкладке *Работа со связями*.

Если сохраняется схема данных, в следующий раз при переходе на вкладку *Работа со связями* при выборе *Работа с базами данных* → *Показать или скрыть* → *Схема данных* можно увидеть ту же структуру таблиц. Это функциональная возможность удобна.

При отказе от сохранения схемы данных, в следующий раз задачей будет повторное создание схемы, включая добавление нужных таблиц и организацию их в окне.

Редактирование связей

При желании изменить или добавить связи, нужно тем же способом вызвать на экран окно *Работа со связями* и выбрать *Работа с базами данных* → *Показать или скрыть* → *Схема данных*.

При сохранении схемы данных, вставленные таблицы появятся автоматически в том виде, в каком их оставили. При работе с таблицами, пока не включенными ни в какие связи, их можно добавить в схему, щелкнув правой кнопкой мыши где-нибудь на свободном пространстве и выбрав команду *Добавить таблицу*.

Если выбрано сохранение схемы данных, можно использовать несколько приемов быстрого возвращения таблиц на экран:

- перетащить таблицы прямо из *Области переходов* на вкладку *Работа со связями*;
- выбрать *Работа со связями* | *Конструктор* → *Связи* → *Все связи* для отображения всех таблиц, включающих связи, созданные ранее;
- добавить таблицу на схему, отметить ее и затем выбрать *Работа со связями* | *Конструктор* → *Связи* → *Прямые связи* для вывода на экран таблиц, связанных с данной таблицей.

Для создания новых связей можно использовать вкладку ленты *Работа со связями*. Созданные связи можно редактировать. Для этого щелкнуть правой кнопкой мыши линию, представляющую связь, и выбрать команду *Изменить связь*. Если в меню нет команды *Изменить связь*, просто не попали на линию

связи. Для удаления связи щелкнуть кнопкой мыши линию связи и выбрать команду *Удалить*.

Обычно редактирование связи заключается в изменении параметров, относящихся к целостности данных на уровне ссылок или ссылочной целостности.

Целостность на уровне ссылок

Как и в реальном мире, отношения налагают некоторые ограничения. В мире БД эти правила называются целостностью данных на уровне ссылок. Все вместе они гарантируют постоянную непротиворечивость связанных данных.

Целостность на уровне ссылок вступает в действие, только, если у связи установлен флажок *Обеспечение целостности данных*.

Программа Access также не разрешит удалить таблицу, участвующую в связи. Нужно сначала удалить связь, используя окно *Работа со связями*, а затем – таблицу.

Пропущенные значения в несвязанных записях

Важно понимать, что есть одна операция, которую можно выполнить, не нарушая целостности данных: создание записи об объекте, не ссылающейся ни на какой другой объект. Это произойдет, если поле объекта останется пустым. Единственная причина, по которой это поле может остаться пустым, – отсутствие записи об объекте в БД или отсутствие приемлемой информации.

Каскадное удаление

Правила целостности решительно останавливают при попытке удалить родительскую запись, на которую ссылаются дочерние записи. Но есть и другой вариант – гораздо более радикальный. Можно выбрать удаление всех связанных дочерних записей при удалении записи-родителя.

Каскадные удаления рискованны. Слишком легко удалить больше записей, чем было намечено, а если это сделать – пути назад нет. Еще хуже то, что команда *Отменить* не сможет помочь отменить это изменение. Поэтому действовать нужно с осторожностью.

Для включения этого варианта при создании связи вы должны установить флажок *Каскадное удаление связанных записей*. Обновить связь можно и позже, установив данный флажок.

Средство *Каскадное удаление связанных записей* – ядерное оружие БД, поэтому необходимо хорошенько подумать, необходимо ли оно. Этот режим позволяет очень легко удалить записи, которые на самом деле нуждались всего лишь в корректировке.

Каскадные обновления

Программа Access также позволяет задать каскадное обновление. Если включить этот режим, установив флажок *Каскадное обновление связанных записей* в диалоговом окне *Изменение связей*, Access копирует любое изменение, сделанное в связанном поле родительской записи, во все дочерние.

Каскадные обновления безопаснее каскадных удалений, но они редко нужны. Поскольку, если следовать правилам хорошего проектирования БД, устанавливается связь, используя столбец с типом данных *Счетчик*. Программа Access не разрешает корректировать значение типа *Счетчик*, и каскадное обновление никогда не понадобится.

Переходы в отношении

Отношения не только помогают вылавливать ошибки, они облегчают просмотр связанных данных. Вот как это действует. Если просматривать таблицу-родитель на листе данных, то можно найти все связанные дочерние записи, щелкнув кнопкой мыши квадратик со знаком "плюс", расположенный у левого края строки (рис. 6.5).

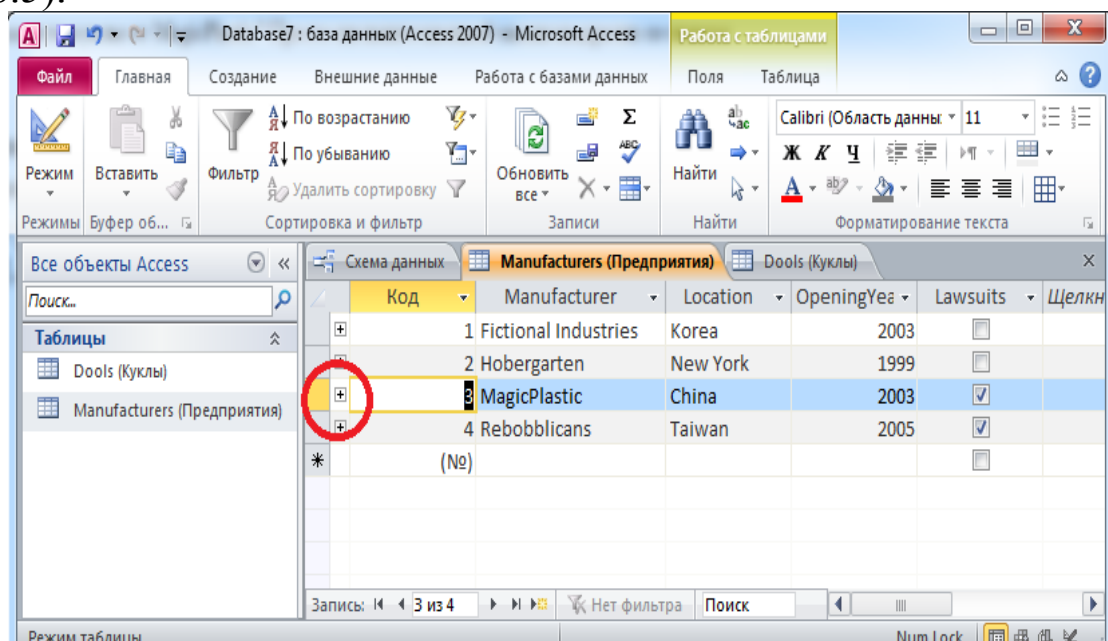


Рис. 6.5. Просмотр связанных записей

Чтобы узнать, какие у вас есть объекты следует просто щелкнуть кнопкой мыши квадратик со знаком "плюс" (обведенный)

Этот щелчок раскрывает подтаблицу, в которой отображаются только связанные записи (рис. 6.6). Подтаблицу можно использовать для редактирования записей прямо в этом окне так, будто выполняется работа на листе данных с полной таблицей. Можно даже добавлять новые записи.

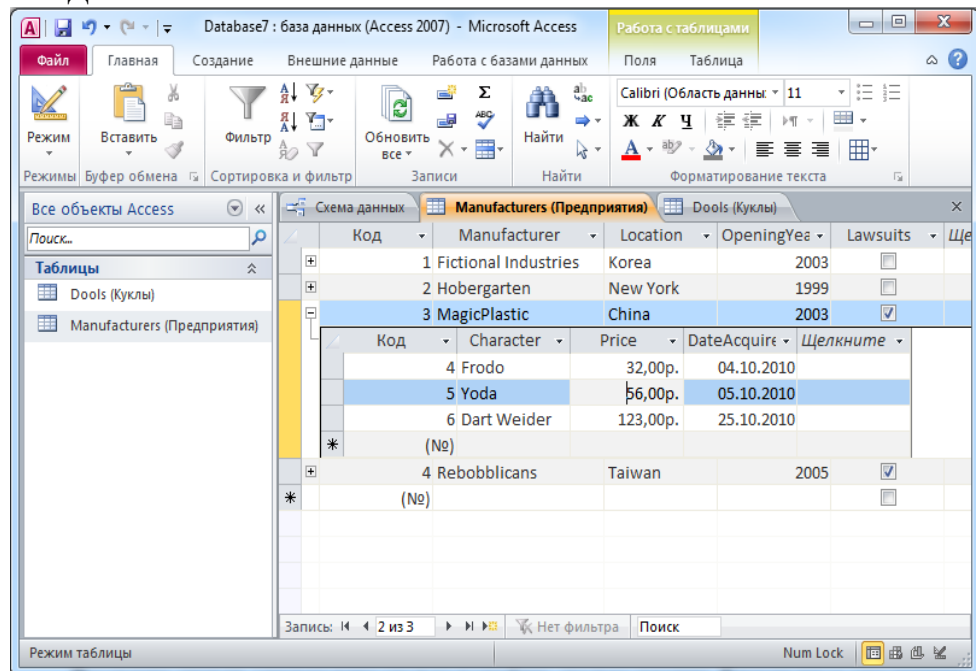


Рис.6.6. Просмотр связанных записей

Можно отрегулировать несколько дополнительных параметров, влияющих на способ отображения в таблице созданных подтаблиц. Для вывода на экран этих параметров нужно перевести таблицу в *Конструктор*. Затем выбрать на ленте *Работа с таблицами* | *Конструктор* → *Показать или скрыть* → *Страница свойств*. *Страница свойств* отображается в правой части окна. На ней есть набор разнообразных параметров, которые применяются к таблице в целом. Далее перечислены те из них, которые относятся к подтаблицам:

- *Имя подтаблицы*. Связанная таблица, применяемая в качестве подтаблицы. Если имеется несколько связанных таблиц, можно выбрать ту, с которой предполагается работать. Или установить значение параметра *Auto*, которое заставит программу Access спросить имя подтаблицы в

следующий раз, когда щелкнут кнопкой мыши квадратик со знаком плюс.

- *Высота подтаблицы.* Задаёт высоту в дюймах, отводимую подтаблице для отображения данных. Если все связанные строки не помещаются в отведенное пространство, придется пользоваться полосой прокрутки. Стандартное значение этого параметра – 0, позволяющее подтаблице занять столько места, сколько ей нужно.
- *Развернутая подтаблица.* Позволяет выбрать вывод свернутых подтаблиц до тех пор, пока не щелкнуто кнопкой мыши по квадратику с плюсом (значение по умолчанию), или задать автоматическое раскрытие подтаблицы при открытии основной таблицы, выбрав значение *Да*.

Поиск в связанных таблицах

Связи облегчают просмотр и редактирование записей, они обычно основываются на значении типа *Счетчик*. При создании подстановки можно представить список констант или предложить список значений из другой таблицы, которые можно использовать.

При создании подстановки, содержащей несколько столбцов, один столбец содержит информацию, которая выводится для пользователя БД, в других столбцах находятся данные, которые предполагается использовать при выборе значения.

Программа Access при переходе к подстановкам ждет добавления списка подстановок, а потом связи. В действительности, когда создается подстановка, использующая таблицу, Access создает связь автоматически. Далее приведены действия, необходимые для создания списка подстановок, связывающего две таблицы:

- Открыть таблицу-потомок в *Конструкторе*.
- Выбрать поле, связывающее ее с таблицей-родителем, в столбце *Тип данных* выбрать вариант *Мастер подстановок*.
- Выбрать переключатель *Объект "столбец подстановки"*, который будет использовать значения из таблицы или запроса и щелкнуть мышью кнопку *Далее*.
- На экране появится список всех таблиц БД за исключением текущей таблицы.
- Выбрать таблицу-родитель и нажать кнопку *Далее*.

- Добавить поле, которое используется для связи, и еще одно, более информативное поле в список *Выбранные поля*. Для продолжения щелкнуть мышью кнопку *Далее*.
- Выбрать поле, применяемое для сортировки списка подстановки, и щелкнуть мышью кнопку *Далее*.
- В следующем окне мастера показывается предварительное представление списка подстановки. Необходимо убедиться в том, что установлен флажок *Скрыть ключевой столбец*, и затем щелкнуть мышью кнопку *Далее*. Несмотря на то, что у поля первичного ключа есть значение, связывающее вместе две таблицы, для пользователя, работающего с БД, оно значит не слишком много. Ему гораздо важнее другое, описательное поле.
- Выбрать название столбца подстановки.
- На последнем этапе можно также выбрать режим, называемый *Разрешить несколько значений*. Если установить этот флажок, в списке отображается флажок рядом с каждым значением, поэтому можно одновременно выбрать несколько элементов списка.
- Щелкнуть мышью кнопку *Готово*.

Теперь программа Access формирует список подстановки для поля и предлагает сохранить таблицу. После этого Access создает связь между двумя таблицами, связанными столбцом подстановки.

Созданная программой Access связь не обеспечивает ссылочной целостности, поскольку программа не знает, соответствуют ли записи этому жесткому стандарту. Если такая возможность кажется угрожающе нестрогой, можно отредактировать связь с помощью вкладки *Схема данных*. Затем щелкнуть правой кнопкой мыши линию связи между ними и выбрать команду *Изменить связь*. В заключение установить флажок *Обеспечение целостности данных* и щелкнуть мышью кнопку *ОК*.

Отношение "один-к-одному"

Отношение или связь "один-к-одному" связывает одну запись таблицы с одной или не связывает ни с одной записью другой таблицы. Иногда этот тип отношения применяется для разбиения таблицы с большим количеством полей на две или несколько меньших таблиц.

Когда связываются два поля, в которых не допускаются дублирующиеся данные и флажок *Обеспечение целостности данных* установлен, Access считает, что создается связь "один-к-одному". Программа помещает цифру 1 на концах линии связи для того, чтобы отличать ее от других типов связей. Отношение "один-к-одному" создается так же, как отношение "один-ко-многим" – перетаскиванием с помощью мыши полей на вкладке *Схема данных*. Единственная разница состоит в том, что в связанных полях обеих таблиц нужно задать запрет совпадений. В этом случае запись одной таблицы может как максимум быть связана с единственной записью в другой таблице (рис.6.7).

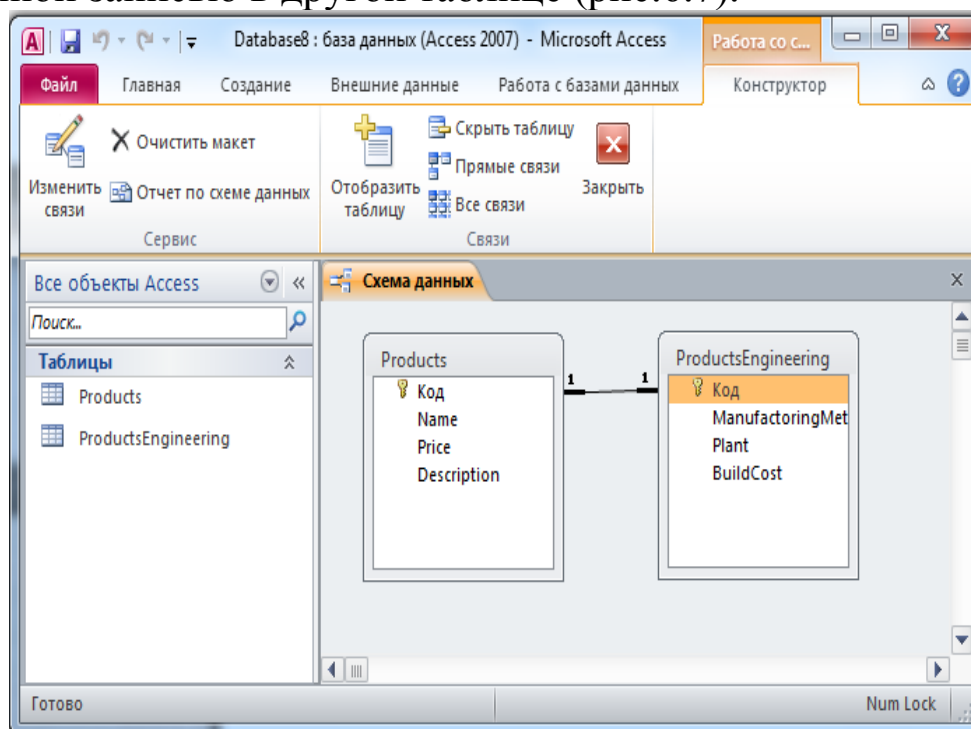


Рис.6.7. Установка связи «один – к – одному»

В поле запрещены совпадения, если оно является первичным ключом таблицы или, если у поля есть индекс, препятствующий появлению дублирующейся информации.

Отношения "один-к-одному" крайне редко применяются в программе Access. Обычно гораздо удобнее использовать скрытие столбцов и запросы, если нужно видеть только отдельные поля таблицы.

Разделение таблицы на две части усложняет проект БД и обычно это делается только, если есть другие причины для разбиения таблицы. Примерами могут служить следующие варианты:

- Две части таблицы необходимо поместить в отдельные БД для того, чтобы разные люди могли копировать их на разные компьютеры и редактировать независимо.
- Если нужно защитить от любопытных глаз уязвимые данные. Один из возможных способов – поместить информацию, которую нужно защитить, в отдельную таблицу и сохранить эту таблицу в другой, более защищенный файл БД.
- Если есть таблица с огромным объемом данных, таких как поле *Вложение* с большими документами. В этом случае можно повысить производительность, если разделить таблицу. Даже можно решить, что лучше поместить половину таблицы в отдельную БД.
- Некоторые данные таблицы необязательны. Вместо того чтобы включать большое количество незаполненных полей, можно выделить их в отдельную таблицу. Когда не нужно включать эту информацию, не придется добавлять запись в связанную таблицу.

Если нет таких ситуаций, можно больше выиграть от создания одной большой таблицы.

Отношение "многие-ко-многим"

Отношение или связь "многие-ко-многим" связывает одну или несколько записей одной таблицы с одной или несколькими записями в другой таблице. Рассмотрим БД, в которой в отдельных таблицах хранятся данные об авторах и книгах. Авторы бестселлеров не останавливаются на одной книге, поэтому нужно иметь возможность связать одного автора с несколькими книгами. Однако иногда авторы объединяются в команду под одним заглавием, поэтому нужно иметь возможность связать одну книгу с несколькими авторами. Аналогичная ситуация возникает, если нужно распределить студентов по курсам, сотрудников по отделам. Связи "многие-ко-многим" довольно распространены, и программа Access предоставляет два способа их обработки: связующие таблицы и многозначные поля.

Связующие таблицы

Связующие таблицы – традиционный метод обработки связей "многие-ко-многим", и их используют повсеместно в мире БД,

включая и программное обеспечение промышленного уровня, такое как *Microsoft SQL Server*. Основная идея состоит в том, что создается дополнительная таблица, у которой единственное назначение – связывание двух таблиц.

Каждая запись в связующей таблице представляет связь, которая соединяет вместе запись каждой таблицы в отношении. В БД с книгами и авторами единственная запись в связующей таблице сопоставляет одного автора с одной книгой. Если один и тот же автор написал три книги, нужно добавить три записи в связующую таблицу. Если два автора работают над одной книгой, потребуется дополнительная запись для связи с каждым новым автором.

Имя связующей таблицы часто состоит из имен двух таблиц, которые она связывает.

Суть связующей таблицы заключается в том, что она формирует два отношения "один-ко-многим", определенные в программе Access. Другими словами, связующая таблица – это таблица-потомок, у которой два родителя. Эти два отношения можно определить на вкладке *Схема данных*, убедившись в том, что заданы правила целостности данных (рис. 6.8).

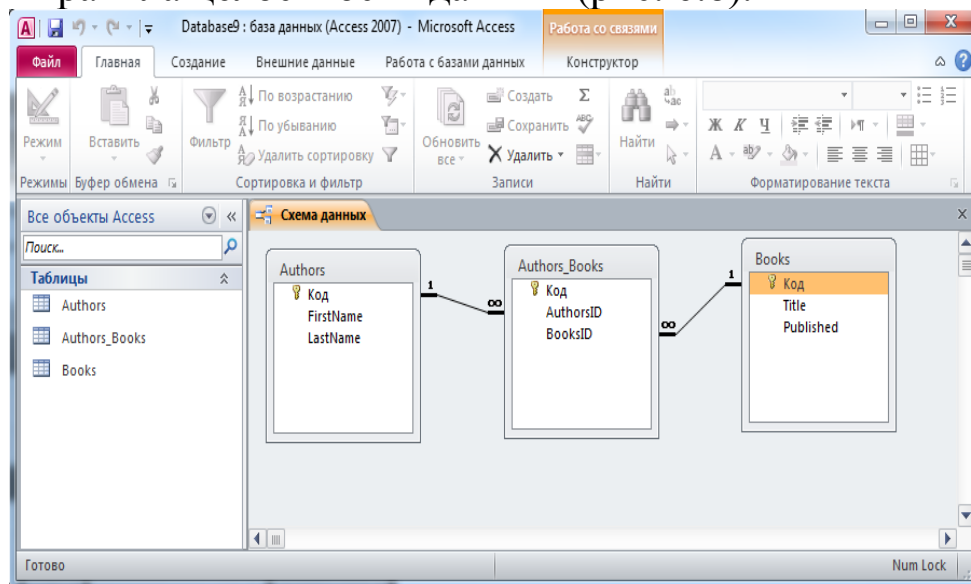


Рис.6.8. Установка связи «многие-ко многим»

Многозначные поля

До появления программы Access 2007 связующие таблицы были единственным средством создания связей "многие-ко-многим". Но для поддержки средств интеграции сервисов *SharePoint* в Access 2007 включена новая функциональная

возможность – многозначные поля. В Access 2010 это возможность сохранилась.

Как следует из названия, многозначное поле может хранить более одного значения. Эта возможность очень удачно решает проблему связей "многие-ко-многим". Идея состоит в настройке связующего поля в таблице-потомке как многозначного поля. Вернемся к примеру с авторами и книгами. При отсутствии связующей таблицы нужно вставить столбец *Код автора* в таблицу с записями о книгах для обозначения каждого автора, написавшего данную книгу.

Для создания поля с несколькими значениями следует использовать подстановку. Эта функциональная возможность выбирается на последней странице мастера *Создание подстановки*. С другой стороны, если уже есть подстановка в поле, необходимо внести небольшое изменение: открыть таблицу в *Конструкторе*, выбрать поле с подстановкой и затем в области *Свойства поля* щелкнуть кнопкой мыши вкладку *Подстановка*, найти свойство *Разрешение нескольких значений* и изменить его значение с *Нет* на *Да*.

Как только задана в поле поддержка множественных значений, нельзя вернуться к варианту поддержки одного значения.

В списке подстановки применяются флажки, поскольку он предназначен для многозначного поля. В одной записи можно выбрать несколько значений, установив флажки нескольких элементов списка.

Многозначные поля доступны, только, если применяется БД нового формата с расширением *accdb*. В файле с расширением *mdb* (БД, созданной программой Access 2003 и еще не преобразованной) нельзя их использовать.

Поля с множественными значениями вызывают проблемы при переносе БД на *SQL Server*, поскольку *SQL Server* не поддерживает их. Следовательно, если есть вероятность совместного использования БД многими пользователями, и в какой-то момент нужно перенести данные в БД более мощной программы *SQL Server*, необходимо избегать полей с множественными значениями.

Печать отношений

Последовательность *Файл → Печать* становится недоступной, когда просматривается вкладка *Схема данных*.

Напечатать непосредственно вкладку *Схема данных* нельзя, но ее можно преобразовать в отчет, представляющий собой специализированный объект БД и позволяющий создать распечатку в любое время.

Для создания отчета о связях таблиц сначала следует расположить все выбранные таблицы на вкладке *Схема данных*, затем выбрать *Работа со связями | Конструктор → Сервис → Отчет по схеме данных*. На экран выводится окно предварительного просмотра, которое похоже в той или иной степени на текущее содержимое вкладки *Схема данных*. Для того чтобы вывести его на печатающее устройство, можно выбрать последовательность *Файл → Печать*.