

ИЗУЧЕНИЕ ТИПОВ ОБЪЕДИНЕНИЯ В БАЗАХ ДАННЫХ В ШКОЛЬНОМ КУРСЕ ИНФОРМАТИКИ

Шамшина Наталья Владимировна,
*старший преподаватель кафедры информатики
Сумского государственного педагогического университета имени А. С. Макаренко,
г. Сумы, Украина/
E-mail: shamichek@ukr.net*

Статья посвящена изучению типов объединения, которые поддерживаются в режиме Конструктора запросов: внутренние объединения – одно-столбцовые и много-столбцовые; внешние объединения – левое и правое; само-объединения; тета-объединения по условию неравенства, перекрестные объединения. Приведены примеры их использования при построении запросов.

Ключевые слова: методика изучения; Access; базы данных; типы объединения; запросы.

STUDYING THE TYPES OF JOINS IN DATABASES IN THE COURSE OF COMPUTER SCIENCE AT SCHOOL

Shamshina Natalya,
*art. Lecturer, Department of Computer Science
Sumy state pedagogical university behalf of A.S. Makarenko,
Sumy, Ukraine.
E-mail: shamichek@ukr.net*

The article is devoted to the study the types of joins that are supported in the mode of Query Designer: internal joins – single-columned and multi-column; external joins – left and right; self-joins; theta-joins with the condition of inequality; cross-joins. Examples of their use when constructing queries are given.

Keywords: method of study; Access; database; types of joins; queries.

Современные базы данных (БД) и системы управления базами данных (СУБД) изучаются на разных уровнях системы образования. Знакомство с реляционными базами данных происходит еще в старших классах средней школы на примере СУБД Access. Навыки работы с БД необходимы ученикам при решении олимпиадных задач по информационным технологиям.

Вопрос организации связей между таблицами и создания объединений является принципиальным для понимания работы реляционной базы данных. В научно-методической литературе традиционно уделяют много внимания изучению типов связей в реляционных БД, семантическому моделированию БД, построению моделей «сущность-связь». Вместе с тем в учебниках трудно найти материал по вопросу использования различных типов объединения записей при конструировании запросов. Поиск сведений о типах объединения в СУБД Access, как правило, приводит к страницам сайтов, посвященных изучению структурированного языка запросов SQL и его синтаксиса.

По результатам анализа актуальных источников по данной теме можно с уверенностью сказать о том, что существует общепринятое мнение: без овладения языком

SQL невозможно понять и научиться применять на практике различные типы объединений в запросах Access. Это обосновано тем, что в основе всех реляционных СУБД лежит прежде всего SQL, а мастера и конструкторы в СУБД Access – это вторичные визуальные средства. Язык SQL более универсален и позволяет создать такие типы объединений, которые не поддерживаются в Конструкторе запросов. Однако Конструктор запросов упрощает процесс создания запросов, позволяет создавать запросы, не используя SQL.

Цель статьи – описать различные типы объединения записей, которые поддерживаются в Конструкторе запросов, и привести примеры их использования для решения практических задач.

«Связь», «объединение» – значение этих слов трудноразлично для школьника, начинающего знакомство с БД. Обратимся к определениям, которые дает Р. Дженнингс – программист, стоящий у истоков СУБД Access, автор специального издания «Использование Microsoft Access 97»: «В теории СУБД известны четыре варианта связей между двумя таблицами, традиционно называемых отношениями (Relationship): один-к-одному, один-ко-многим, многие-к-одному, многие-ко-многим» [1, с. 117].

Связи указывают на то, как первая таблица (главная) связана со второй таблицей (подчиненной) и какое количество записей может соответствовать друг другу в этих таблицах. В главную таблицу записи вводятся в первую очередь, в подчиненную – после. Отношение «многие-ко-многим» осуществляется посредством еще одной связующей таблицы – таблицы ассоциаций.

Связи реализуются за счет одинаковых значений в полях связи, естественно, тип и подтип полей должны совпадать. Тип отношения определяется программой с учетом уникальных индексов в полях связи, для ключевых полей индекс устанавливается автоматически. В таблицах со стороны «один» значения в поле связи уникальны, не повторяются, как правило, это поле является первичным ключом. В таблицах со стороны «многие» значения в поле связи могут повторяться.

Объединением (Join) называется операция, при которой происходит сопоставление и комбинирование значений в общих полях связанных таблиц, являющихся источником данных в форме, отчете или запросе.

Связи между таблицами устанавливаются на Схеме данных. Они отображаются линиями, которые соединяют имена полей связи в таблицах. Существующие связи действуют в последующих операциях обработки и вывода данных – при создании форм, отчетов, запросов. Важную роль при этом играет тип объединения.

Тип объединения задает способ просмотра связанных записей на основе заданного отношения. Для просмотра данных одновременно двух таблиц в форме, отчете, запросе выбирают один из трех типов объединения:

– равное или внутреннее объединение – объединение только тех записей, в которых связанные поля обеих таблиц совпадают;

– внешнее левое объединение – объединение ВСЕХ записей из первой таблицы и только тех записей из второй таблицы, в которых связанные поля совпадают;

– внешнее правое объединение – объединение ВСЕХ записей из второй таблицы и только тех записей из первой таблицы, в которых связанные поля совпадают.

Внутреннее объединение отображает записи в объединенных полях как одну запись. Если не найдено соответствующего значения в поле связи связанной таблицы, данные не отображаются вообще. Можно сказать, что отображаются записи-пары. Таким образом количество записей, которые может видеть пользователь в запросе, форме, отчете, равно количеству записей в связанной таблице. Для таблиц, которые связаны отношением «один-ко-многим» будут видны повторы части записи со стороны таблицы «один». Это наиболее распространенный тип объединения, установленный по умолчанию.

Внешнее объединение позволяет отображать поля всех записей таблицы независимо от существования связанных записей в объединенной таблице. Таким образом количество записей больше, чем при равном внутреннем объединении. Применяют левые и правые внешние объединения. Названия «левое» и «правое» возникли в соответствии с традицией размещать главную таблицу на схеме данных слева, а связанную – справа [1, с. 266]. Поэтому при внешнем левом объединении отображаются все записи главной таблицы: записи-пары и записи без пары. Внешнее левое объединение используют для удобного добавления записей в подчиненной таблице. При внешнем правом объединении отображаются все записи подчиненной таблицы и связанные с ними из главной таблицы, то есть пары записей и оторванные записи «сироты». Внешнее правое объединение используют для поиска и удаления записей – «сирот». При наличии записей – «сирот» невозможно подключить обеспечение целостности данных, что является обязательным для рабочих баз данных [2].

Тип объединения можно указать через контекстное меню линии связи. Чаще всего необходимость изменить тип объединения возникает при конструировании запроса. В Access для создания запросов можно использовать либо бланк запроса по образцу (Query by Example – QBE), либо режим SQL. Бланк запроса первоначально был создан, чтобы позволить пользователям компьютерных приложений баз данных, незнакомых с языками программирования, извлекать и отображать требуемые данные [1, с. 206]. Access переводит в операторы SQL запросы, созданные с помощью графического интерфейса в Конструкторе запросов.

В режиме конструктора запросов поддерживается 5 видов объединения, которые подразделяются на типы [3]:

- внутреннее объединение – одно-столбцовое и много-столбцовое;
- внешнее объединение – левое и правое;
- само-объединение;
- перекрестное объединение;
- тета-объединение по условию неравенства.

При конструировании запроса следует внимательно указывать источники данных для результирующего набора записей и учитывать следующее:

- если добавить в Конструктор запросов таблицы, непосредственно связанные на схеме данных, автоматически действует существующая связь;
- если таблицы не связаны непосредственно, нужно добавить всю цепочку связанных таблиц в качестве источника данных;
- если добавить в Конструктор запросов несвязанные таблицы – получим декартово произведение, или перекрестное объединение, при котором к каждой записи первой таблицы с N1 записями подставляется каждая запись второй таблицы с N2 записями, общее количество полученных записей в запросе равно $N1 \times N2$.

В Конструкторе запросов создают объединение перетягиванием поля связи из одной таблицы на соответствующее поле в другой таблице. Сопоставление записей осуществляется по одинаковым значениям полей, тип которых должен совпадать. В отличие от выполнения операции на Схеме данных, тип отношения не определяется, обеспечение целостности данных не подключается.

Одно-столбцовое внутреннее объединение основано на связи одного поля-столбца в каждой таблице. Много-столбцовое внутреннее объединение основано на связях нескольких пар столбцов. Используется для отбора записей, в которых значения объединенных полей совпадают.

Само-объединение связывает поля одной таблицы и отображает только те записи, в которых значения этих полей совпадают. Для создания само-объединения в Конструкторе запросов таблицу добавляют дважды, дубликату автоматически назначается псевдоним. Затем указывают объединение полей. Само-объединения применяют чаще в

запросах на создание новой таблицы. В запросах на выборку более надежно использовать условие на значение.

Перекрестное объединение (декартово произведение) не отображается линиями, используется для генерации комбинаций записей двух таблиц.

Тета-объединения связывают данные с помощью операторов сравнения, отличных от оператора равенства. Тета-объединения не отображаются линиями и применяются в запросах для отбора записей, которым нужно отношение особого типа. Если в столбец бланка запроса включить условие неравенства значений полей, получим тета-объединение.

Необходимо подчеркнуть, что в Конструкторе запросов создаются объединения между полями с неопределенным типом отношения, следовательно, сопоставление записей из двух таблиц может быть неоднозначным. Если встречаются повторяющиеся значения в обоих полях связи – результирующий набор запроса имеет все возможные комбинации. Это приводит к неправильным результатам в запросах на выборку. Однако это оказывается полезным при решении задач, в которых нужно на основе данных таблицы сформировать новую таблицу с большим количеством записей.

Рассмотрим примеры конструирования с помощью запроса набора записей, который является комбинированием значений одного поля заданной таблицы. Это могут быть названия спортивных команд, соревнующихся между собой, или же названия торгующих между собой стран.

Допустим, имеем 15 названий стран в таблице «Страны». Используем перекрестное объединение и тета-объединение по условию неравенства (рис. 1). Сначала добавляем таблицу «Страны» дважды в окно Конструктора запроса. «Страны_1» – псевдоним копии таблицы. Далее в бланке запроса указываем поля для вывода. Получим декартово произведение $15 \times 15 = 225$ записей. Среди комбинаций есть 15 лишних записей, когда одно и то же название сопоставляется (например, «Белоруссия-Белоруссия»). Лишними также в данном случае являются записи-повторы комбинаций, в которых названия стран переставлены (например, «Украина-Белоруссия», «Белоруссия-Украина»). Для избавления от лишних записей в столбце поля *Страны.название* указываем выражение для условия: $< [Страны_1]![Название]$. Получим 105 комбинаций названий стран, которые не повторяются.

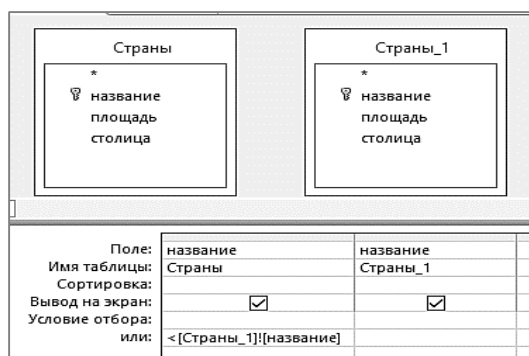


Рис. 1. Тета-объединение

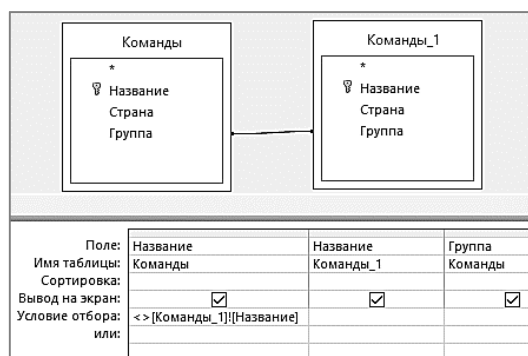


Рис. 2. Само-объединение

На рис. 2 приведен пример использования само-объединения при конструировании запроса с набором комбинаций названий спортивных команд, который является решением следующего олимпиадного задания: «создайте запрос, который на основе представленной в таблице *Команды* информации о футбольных командах генерирует все матчи группового этапа; в групповом этапе каждая команда должна дважды сыграть с каждой другой командой своей группы, причем один раз дома, а один на выезде; каждая группа состоит из четырех команд». В таблице *Команды* 48 названий команд из

12 групп от А до L из разных стран, по 4 команды в группе. Создаем само-объединения по полю *Группа*, получим $4 \times 4 = 16$ комбинаций названий команд в рамках каждой группы, вместе $16 \times 12 = 192$. С помощью условия неравенства названий: \diamond [Команды_1]! [Название] – избавляемся от комбинаций из одинаковых названий (их по 4 в каждой группе, $4 \times 12 = 48$). Получим $192 - 48 = 144$ комбинации результирующего набора записей по 12 матчам группового этапа для 12 групп.

Точки на концах линии связи означают, что объединение создано между полями с неопределенным типом отношения. Этот пример иллюстрирует тот факт, что при наличии повторений в значениях полей связи в обеих таблицах результирующий набор записей содержит все возможные комбинации их значений.

Выводы. Анализ различных типов объединения дает основания сделать следующие выводы относительно их использования в запросах QBE: наиболее востребованные на практике типы объединения в БД с легкостью применяются в Конструкторе и дают правильные результаты.

В Конструкторе запросов можно использовать внутренние и внешние объединения, само-объединение, перекрестное объединение, тета-объединение по условию неравенства. Изучение типов объединения в БД в школьном курсе информатики, решение задач на их использование возможно без освоения синтаксиса SQL.

Необходимо с большой осторожностью использовать объединения с неопределенным типом отношения в запросах на выборку, обязательно контролировать наличие повторяющихся значений и количество записей, которые получились в результате выполнения запроса.

Изучение типов объединения развивает логическое мышление, помогает усвоить и закрепить навыки конструирования запросов, приучает к контролю возможных ошибок в результирующем наборе записей, мотивирует к нешаблонному креативному подходу при решении практических задач.

Список литературы

1. Дженнингс Р., Использование Microsoft Access 97 : спец. изд., пер. с англ. – 2-е изд. – К., М., СПб : Вильямс, 1998. – 944 с.
2. Шамшина Н. В., Об особенностях сохранения информации в базах данных // Фізико-математична освіта: науковий журнал. – 2016. – Випуск 4(10). – С. 148 – 151.
3. Join tables and queries, Applies To: Access 2016 Access 2013 Access 2010 Access 2007. – URL: <https://support.office.com/en-us/article/join-tables-and-queries-3f5838bd-24a0-4832-9bc1-07061a1478f6?ui=en-US&rs=en-US&ad=US> (дата обращения: 10.02.2018).