

Математика не для ЕГЭ

Е. К. Белый

Введение
в Microsoft Access

Учебное электронное пособие
для учащихся средних школ

Петрозаводск
Издательство ПетрГУ
2020

УДК 004.65

ББК 32.973

Б439

Рецензенты:

Н. А. Будникова, кандидат физико-математических наук, доцент кафедры информатики и математического обеспечения ПетрГУ;

А. А. Морозов, младший научный сотрудник отдела институционального развития регионов института экономики КарНЦ РАН

Белый, Евгений Константинович.

Б439 Введение в Microsoft Access [Электронный ресурс] : учебное электронное пособие для учащихся средних школ / Е. К. Белый ; М-во науки и высшего образования Рос. Федерации, Федер. гос. бюджет. образоват. учреждение высш. образования Петрозавод. гос. ун-т. — (Математика не для ЕГЭ). — Электрон. дан. — Петрозаводск : Издательство ПетрГУ, 2020. — 1 электрон. опт. диск ; (CD-R) ; 12 см. — Систем. требования : PC, MAC с процессором Intel 1,3 ГГц и выше ; Windows, MAC OSX ; 256 Мб ; видеосистема : разрешение экрана 800X600 и выше ; графический ускоритель (опционально) ; мышь или другое аналогичное устройство. — Загл. с этикетки диска.

ISBN 978-5-8021-3633-1

Учебное пособие ориентировано на учащихся средних школ и учителей информатики.

ISBN 978-5-8021-3633-1

УДК 004.65

ББК 32.973

© Белый Е. К., 2020

Содержание

Предисловие	5
Глава 1. Сущности, таблицы и запросы	7
§ 1.1. Концептуальная модель	7
§ 1.2. Таблица «Ученики»	12
§ 1.3. Таблица «Преподаватели»	23
§ 1.4. Таблица «Кружки»	30
§ 1.5. Связи	33
§ 1.6. Запросы	36
Глава 2. Формы и макросы	43
§ 2.1. Форма «Ученики»	43
§ 2.2. Форма «Список учеников»	52
§ 2.3. Объекты, свойства, события и методы	57
§ 2.4. Форма «Поиск учеников»	69
§ 2.5. Формы «для преподавателей»	74
§ 2.6. Формы «для кружков»	76
§ 2.7. Подчиненные формы	80
Глава 3. Кнопочная форма	89
§ 3.1. Диспетчер кнопочных форм	89
§ 3.2. Один полезный макрос	93
§ 3.3. Страницы кнопочной формы	95

§ 3.4. Параметры запуска	101
§ 3.5. Перекрывание окон	102
§ 3.6. Редактирование кнопочной формы	105
Глава 4. Отчеты	113
§ 4.1. Отчет «Весь список»	113
§ 4.2. Отчет «Ученики по классам»	118
§ 4.3. Отчет «Возраст учащихся»	125
§ 4.4. Отчет «Ученики по кружкам»	140
§ 4.5. Отчет «Количество учеников»	148
§ 4.6. Редактирование кнопочной формы	157
Отладка приложения	160
Историческая справка	168
Список литературы	173

*Программа выполняет не то,
что хочет программист,
а то, что он написал.*

IT-афоризм

Предисловие

⇒7 Дорогой читатель! Эта книга позволит получить основные навыки создания баз данных в Microsoft Access. Прежде чем приступить к работе, следует определиться с выбором версии Access. Если мы хотим, чтобы наше приложение работало только у нас, можно взять последнюю. Дело в том, что версии Access совместимы снизу вверх, но никак не наоборот. Приложение Access 2003 может работать в среде Access 10, но мы не сможем открыть приложение Access 10, если на нашем компьютере установлен Access 2003. Почему бы тогда не остановиться на версии 2003? Начиная с 7-й произошли существенные изменения, и мы не хотим игнорировать этот факт: появился эффективный механизм присоединения к записям фотографий и других объектов; вместо модулей стали использовать внедренные макросы (что это такое, вы скоро узнаете); и наконец, принципиально изменилось оформление окна базы данных. Поэтому мы остановились на Access 10.

Сейчас нет недостатка в литературе по базам данных. В частности, для первого знакомства с Access можно рекомендовать [3], для серьезной работы – [1], для основательного изучения теории – [2].

Автор настоящей книги исходил из того, что лучший способ научиться программировать – самому написать программу. Приложение Access читателю предлагается создать совместно с автором. Это будет база данных, отражающая работу школьных кружков. Для ее разработки мы ограничимся минимальными средствами. В частности, не будем использовать модули, написанные на встроенном языке VBA – Visual Basic for Applications. Зато подробно проиллюстрируем стандартные действия, которые выполняются в процессе разработки баз данных. Пособие ориентировано на практическую составляющую работы с Access. Автор, насколько это возможно, избегал углубления в теоретические вопросы. Хочется надеяться, что книга будет полезна широкому кругу читателей. Как и прежде, замечания и предложения вы можете направлять по одному из адресов:

belyi@petsru.ru или kurs_belyi@mail.ru.

Евгений Белый

Декабрь 2019

Глава 1. Сущности, таблицы и запросы

§ 1.1. Концептуальная модель

5 ⇔ 12 Любой проект начинается с концептуальной модели. Действительно, прежде чем приступить к большой работе, надо понять, **что следует сделать**. Слово «концепция» (лат. *conceptio*) означает «система понимания». Концептуальная модель – это описание множества основных понятий и связей между ними. В теории баз данных такие понятия называют **основными сущностями** [2, с. 41]. Мы выделим три основные сущности: ученики, преподаватели и кружки. На рис. 1 они обозначены

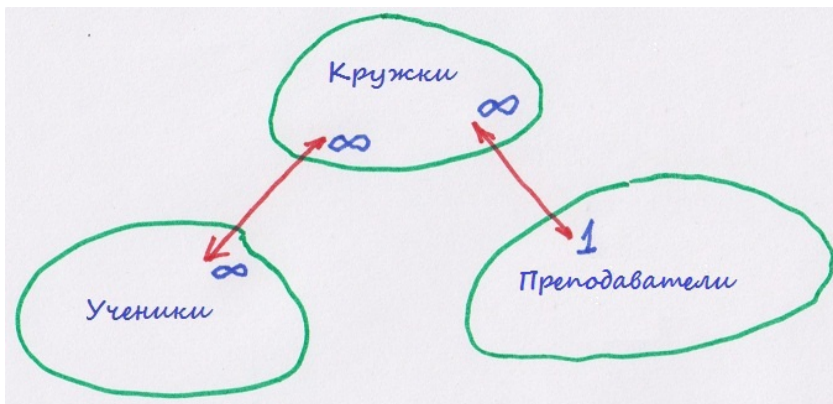


Рис. 1. Концептуальная модель

замкнутыми областями. Стрелочки показывают связи между сущностями. Два символа «∞» у концов стрелки слева означают, что в кружке может быть сколько угодно учеников и что один ученик может заниматься в любом количестве кружков. На самом деле здесь будут иметь место естественные ограничения (во всяком случае ни один школьник не может заниматься в бесконечном количестве кружков). Однако ограничения устанавливаем не мы. Символы «∞» и «1» у концов стрелки справа говорят: занятия в каждой кружке ведет один преподаватель, но преподаватель может работать в нескольких кружках.

Сущность – это то, о чем мы будем хранить данные. Данные должны отражать интересующие нас свойства сущности. У каждой сущности есть набор характерных для нее свойств. Например, сущность «Автомобиль» имеет свойство «Пробег». Для каждого автомобиля, представителя этой сущности, свойство «Пробег» принимает конкретное значение. «Образование» может быть свойством сущности «Люди», но не применимо к автомобилю. Зато свойство автомобиля «Пробег» не может характеризовать человека. В базе данных вся информация о каждом представителе сущности хранится как **запись**. Запись разбита на поля, отражающие различные характеристики сущности. Таким образом, единица данных, образующих запись, – **поле**.

Каждое поле должно иметь имя, позволяющее отличать его от других полей. Поле содержит данные одного и того же типа для каждого представителя сущности. В нашей базе данных запись о представителе сущности «Ученик» будет хранить его фамилию и имя в поле «ФИ» как текстовую строку, а номер класса – в поле «Класс» как целое число от 1 до 11. Теперь определим данные, с которыми мы собираемся работать. Для этого выделим представляющие для нас интерес свойства каждой сущности.

1) Ученики.

Для учеников зарезервируем поля, хранящие фамилию, имя, дату рождения, номер и букву класса, телефонный номер и комментарии.

№	Поле	Описание
1	ФИ	Фамилия и имя
2	ДР	Дата рождения: день, месяц и год
3	Класс	Номер от 1 до 11
4	Буква	Буква после номера класса
5	Телефон	Номер телефона
6	Комментарии	Дополнительная информация

В процессе работы с базой может оказаться, что мы выбрали не самый удачный набор свойств. Ничего страшного. Разработка базы данных – творческий процесс,

и лучшее решение может прийти не сразу. Более того, как бы досконально не очертили мы набор свойств, будут возникать ситуации, когда данные на входе не вписываются в нашу модель. Именно поэтому мы добавили в таблицу поле «Комментарии», предназначенное для хранения произвольной текстовой строки. Если проблемы соответствия данных на входе принятой модели возникают слишком часто, то концептуальная модель нуждается в доработке.

2) Преподаватели.

Первое поле назовем «ФИО». Оно будет хранить фамилию, имя и отчество, т. е. полное имя преподавателя. Иногда для фамилии, имени и отчества выделяют три отдельных поля, но такой подход может привести к непредвиденным осложнениям, например, если ваш кружок будет вести Абу Рейхан Мухаммед ибн Ахмед аль-Бируни. Конечно, это шутка: персидский ученый аль-Бируни жил в XI в. н. э. Но действительно, полные имена представителей некоторых народов заметно отличаются от наших. А потому полное имя лучше хранить в одном поле. Вести кружок может не только школьный учитель, но и мастер шефствующего над школой завода, кто-то из родителей учеников и т. д. Вероятно, представляет интерес должность преподавателя по месту основной работы, его образование, контакты (в данном случае телефон). Можно

добавить электронный адрес.

№	Поле	Описание
1	ФИО	Фамилия, имя, отчество
2	ДР	Дата рождения: день, месяц и год
3	Образование	Высшее, среднее специальное и т. д.
4	Телефон	Номер телефона
5	Должность	Должность по месту работы
6	Комментарии	Дополнительная информация

3) Кружки.

Здесь, помимо названия кружка и ФИО преподавателя, важно планируемое количество часов занятий. Если название кружка не полностью раскрывает его цели и задачи, то подробности можно записать в поле «Комментарии».

№	Поле	Описание
1	Название	Название кружка
2	Преподаватель	ФИО преподавателя
3	Часы	Количество часов за год
4	Комментарии	Дополнительная информация

§ 1.2. Таблица «Ученики»

7 ⇔ 23 Прежде всего создадим папку, в которой будет храниться файл базы данных. Назовем эту папку «Кружки». Откроем папку и щелкнем в ее области правой клавишей мышки. В открывшемся меню выберем: Создать -> Приложение Microsoft Office Access (рис. 2а). В папке появится файл с именем «Приложение Microsoft

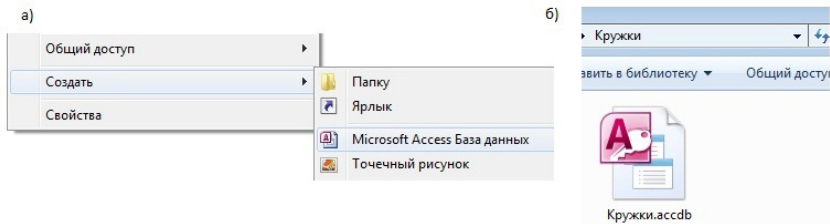


Рис. 2. Создание приложения

Office Access» и расширением accdb. Переименуем файл. Назовем его «Кружки» (рис. 2б). Разумеется, расширение останется без изменения. Откроем файл двойным щелчком левой клавиши мыши. На экране появится окно базы данных (рис. 3). Обратите внимание: слева в разделе «Все объекты Access» пока пусто, потому что мы еще не создали ни одного объекта.

Создание базы данных обычно начинают с таблиц. Каждой из перечисленных в предыдущем разделе сущностей

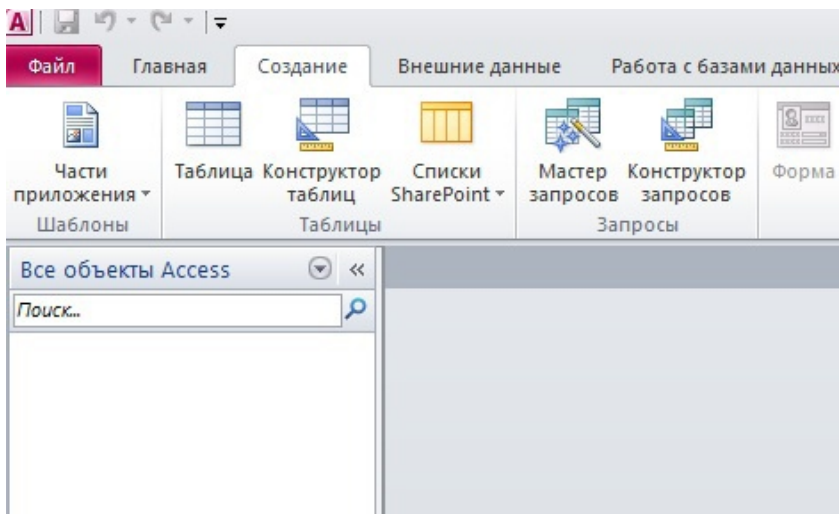


Рис. 3. Окно базы данных

будет соответствовать таблица. При этом запись соответствует строка таблицы, а полю – столбец. На логическом уровне удобно считать, что данные хранятся в таблицах и располагаются в памяти компьютера последовательно: запись за записью, поле за полем. Все записи таблицы имеют одинаковую длину (в байтах), что позволяет избежать лишних ссылок и обеспечивает быстрый доступ к нужной записи. Аналогично одноименные поля в записях имеют одинаковую длину и принадлежат к одному типу данных. Что такое тип данных, знают все, кому приходилось иметь дело хоть с одним из языков программирования. Типы

данных, которые может предложить нам Access, мы разберем по ходу создания таблиц.

Для создания таблицы щелкнем по пункту верхнего меню «Создание», затем по пиктограмме «Конструктор таблиц». На экране появится окно конструктора (рис. 4).

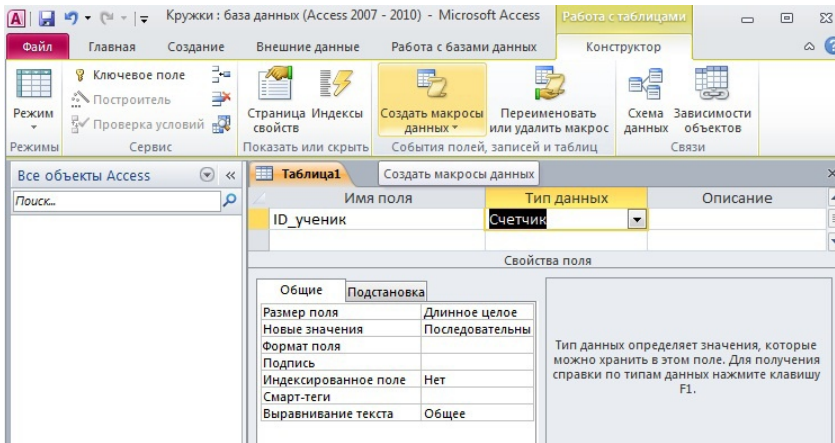


Рис. 4. Конструктор таблиц

В этом окне мы можем создавать и редактировать структуру таблицы. Окно разбито на три колонки: «Имя поля», «Тип данных» и «Описание». Первые две колонки заполняются обязательно, в колонку «Описание» вы можете заносить любую информацию, поясняющую содержание ее полей. Мы оставим заполнение этой колонки на усмотрение читателя. Ниже слева расположены свойства полей, которые вы можете изменять по мере необходимости.

Поля отражают выделенные нами в предыдущем пункте свойства сущностей. Первая таблица будет хранить данные о сущности «Ученик».

1. ID_ученик. Мы добавили одно, не предусмотренное ранее, поле «ID_ученик» (см. рис. 4). Во второй колонке развернем меню и выберем тип данных «Счетчик». Поле типа «Счетчик» в процессе работы с таблицей заполняется автоматически и хранит код записи. Если не менять принятые по умолчанию свойства поля, то для первой записи значение «ID_ученик» будет 1 и далее для каждой следующей записи будет увеличиваться на 1. Таким образом, никогда не появится двух записей с одинаковым кодом. Зачем нужно это поле, скоро будет понятно.

2. ФИ. На следующей строке зададим поле «ФИ», хранящее фамилию и имя ученика (рис. 5). Тип данных выберем текстовый. Здесь требуется внести некоторые уточнения в свойства поля. Размер поля по умолчанию может оказаться слишком большим. Установим 40. В строке «Обязательное поле» установим «ДА». Это значит, что система откажется сохранять запись, в которой не заполнено это поле. Действительно, иначе запись просто бесполезна. Может возникнуть вопрос: а не сделать ли все поля обязательными? Не стоит! Часто, создавая запись, вводят не все данные, и остальные поля заполняются в дальнейшем

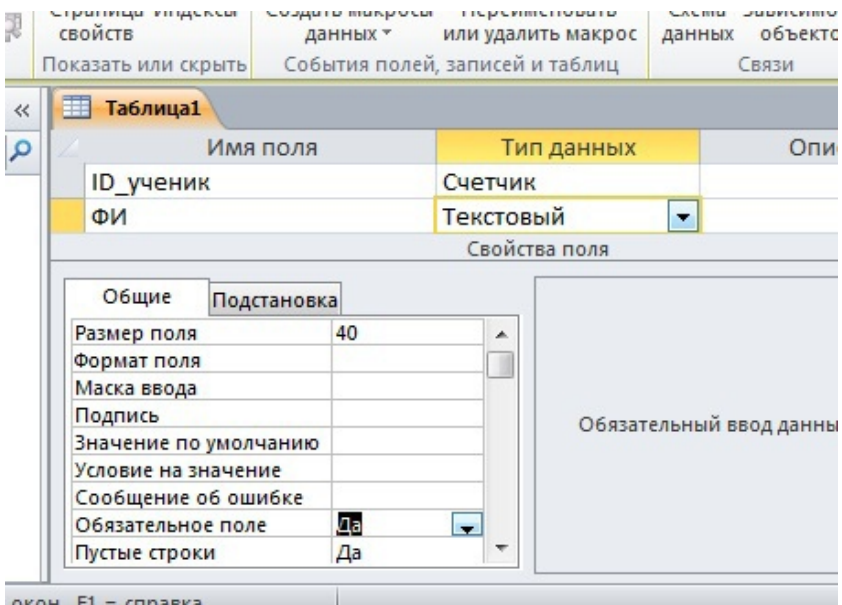


Рис. 5. Первые два поля таблицы

по мере поступления новой информации. Зададим следующие пять полей (рис. 6).

3. ДР. Поле хранит дату рождения. Выберем тип данных «Дата/время». Он позволяет хранить не только день, месяц и год, но и часы, минуты и секунды. Пожалуй, это слишком! В разделе «Свойства поля» уточним формат. Для этого в строке «Формат поля» установим значение «Длинный формат даты».

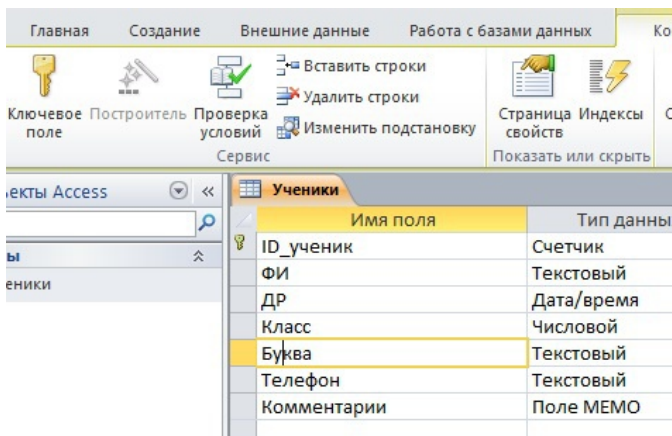


Рис. 6. Все поля

4. Класс. В четвертую строку таблицы конструктора введем имя поля «Класс». Выберем тип поля «Числовой». Здесь тип также нуждается в уточнении. По умолчанию размер поля – «Длинное целое». Это число, принадлежащее интервалу от $-2\ 147\ 483\ 648$ до $+2\ 147\ 483\ 647$. Таких классов не бывает! Откроем в строке «Размер поля» меню и выберем пункт «Байт». Если бы мы этого не сделали, ничего страшного не случилось бы. Однако, если вы хотите стать программистом, вы должны воспитать в себе привычку экономить память. Уместно использовать еще одно свойство числового поля – «Условие на значение». Запишем в этой строке: ≥ 1 And ≤ 11 . Это значит, номер класса должен находиться на интервале от 1 до 11. Если

вы попробуете ввести какое-либо иное значение, система выдаст сообщение об ошибке.

5. Буква. Класс полностью определяется номером и буквой. Может возникнуть вопрос: почему бы нам сразу не задать символьное поле и писать просто 9а, 10б и т. д.? Но тогда мы не смогли бы делать выборки по всем 9-м, 10-м и прочим классам. Поэтому оставим для буквы отдельное текстовое поле, которое так и назовем – «Буква». Длину этого поля можно установить равной 1. Тут есть подводные камни. Сравните два значения поля: «а» и «а». Вы скажете, что они совпадают. А вот и нет! В первом введена английская «а», а во втором – русская. Для системы это разные символы, поскольку им соответствуют разные коды. Такие ситуации непременно возникнут при заполнении базы данных, и у нас будут появляться «лишние» классы. Чтобы избежать подобных неприятностей, в строке поля «Буква» развернем меню и выберем тип данных «Мастер подстановок». На самом деле это никакой не тип данных, а механизм, который позволяет выбирать значение поля из списка. При нажатии на «Мастер подстановок» начнется диалог с «мастером». Установим переключатель в положение «Будет введен фиксированный набор...» (рис. 7) . «Далее». Введем все возможные значения поля «Буква» (рис. 8). «Далее». Мастер попросит

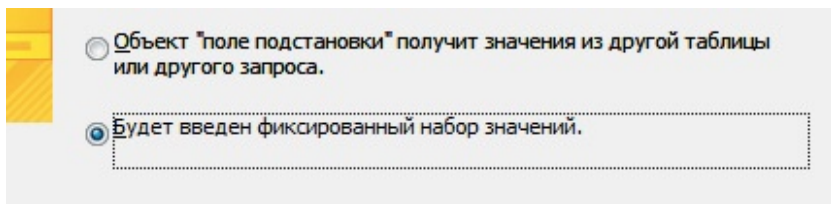


Рис. 7. Выбор фиксированного набора значений

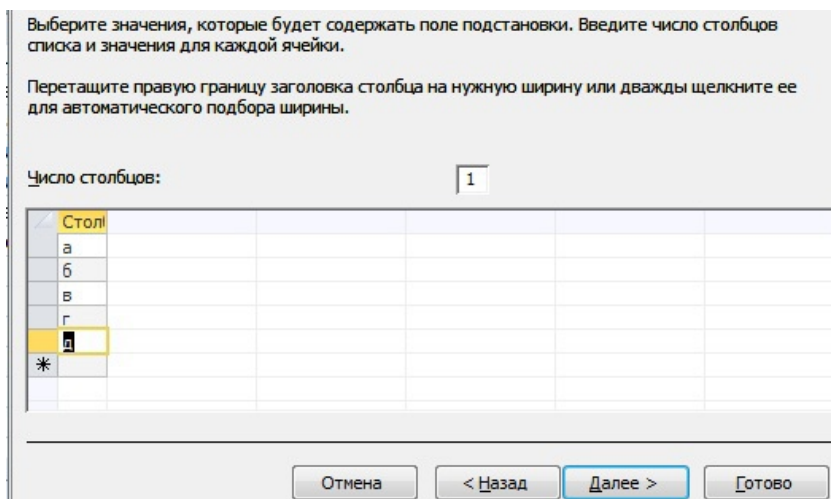


Рис. 8. Фиксированный набор значений

задать имя поля. Оставим прежнее – «Буква». «Готово».

6. Телефон. Для номера телефона выберем тип данных «Текстовый». Размер поля – 12. Как показано на рис. 9, установим маску ввода, т. е. набор символов, задающий формат вводимой информации. Символ «#» означает, что на первом месте в текстовой строке могут стоять «+», «-»,

Размер поля	12
Формат поля	
Маска ввода	#9\999\999\99\99
П...	

Рис. 9. Формат номера телефона

цифра или пробел; символ «9» – на этом месте может стоять только цифра или пробел; перед символами, которые не вводятся, но должны стоять на фиксированных местах, ставится знак «\».

7. Комментарии. Выберем тип данных «Поле МЕМО». Последнее означает, что поле может хранить текстовую строку произвольной длины. Сейчас читатель имеет полное право возмутиться. Ведь выше было сказано, что каждое поле таблицы имеет фиксированную длину. На самом деле поле МЕМО хранит только ссылку на текстовую строку, а сама строка располагается вне таблицы. Поэтому поля МЕМО нежелательно использовать для поиска и сортировки. Переход на строку происходит по ссылке, а значит, эти операции будут выполняться гораздо медленнее.

Похоже, мы сформировали структуру таблицы, однако упустили один важный момент. Если сейчас закрыть окно конструктора с сохранением результата, система напомнит нам, что мы не задали **ключевое поле**. За нами остается право проигнорировать напоминание, но, как правило,

ключевое поле лучше все-таки иметь. Ключевым называют поле, которое однозначно идентифицирует запись. В принципе, ключевым можно сделать любое поле. Но, если, например, мы возьмем в качестве ключевого поле «ФИ», система откажется сохранить запись об Иванове Ване, когда в таблице есть уже один Иванов Ваня. Заметим, в школе не так уж редко встречаются ученики с совпадающими фамилиями и именами. Ключом может быть и совокупность полей. Однако последнее не всегда решает проблему. Так, в базе данных большой больницы могут встретиться два Иванова Ивана Ивановича с совпадающими датами рождения. А перепутать двух больных мы уж точно не имеем права. Вот почему создают специальные поля с кодом записи. Мы с этой целью добавили к полям, хранящим данные о сущности, поле «ID_ученик». Оно дает каждому ученику уникальный код. Сделаем его ключевым. Для этого установим указатель на строке «ID_ученик» и щелкнем по изображению ключа с подписью «Ключевое поле» в левом верхнем углу окна базы данных. Слева от имени поля «ID_ученик» появится изображение ключика. Если теперь читатель еще раз взглянет на рис. 6 (с. 17), он заметит, что мы заранее сделали это поле ключевым. Но мы оставили разговор о ключевых полях «на десерт».

Закроем окно конструктора, т. е. нажмем на крестик в правом верхнем его углу. Появится диалоговое окно «Сохранить изменения макета или структуры объекта?». Ответим: «ДА». В следующем окне нам предложат дать имя таблице. Назовем ее «Ученики». Слева в области «Все объекты Access» появилась первая таблица с присвоенным нами именем. Если в дальнейшем у нас возникнет необходимость отредактировать структуру таблицы (например, добавить еще одно поле), мы можем щелкнуть правой клавишей мышки по имени таблицы и в открывшемся меню выбрать пункт «Конструктор». На экране появится окно конструктора.

В процессе разработки приложения нужен реальный или похожий на реальный набор данных. Щелчком по имени «Ученики» в области «Все объекты Access» откроем таблицу для заполнения. Мы ввели 20 записей (рис. 10). Если введем больше, будет только лучше. При вводе данных следует обратить внимание на следующие моменты:

1. Хотя дата рождения представляется в длинном формате, вводить ее удобней в кратком. Например, вместо «1 января 2005» можно ввести «1.1.5» или «1/1/5».
2. Значение поля «Буква» можно выбрать из меню, но быстрее ввести с клавиатуры.
3. Поле «Комментарии» не стоит заполнять сразу.

ID_ученик	ФИ	ДР	Класс	Буква	Телефон	Комментарий
1	Кошкин Вася	1 января 2006 г.	9 б		+7(921)002-12-12	
2	Тарасенко Гриша	12 марта 2004 г.	10 а		+7(960)003-23-23	
3	Лева Мужин	2 декабря 2005 г.	10 а		+7(921)003-13-13	
4	Кузина Катя	5 мая 2005 г.	10 б		+7(911)002-31-23	
5	Бучиков Вена	13 ноября 2005 г.	9 а		+7(921)001-21-21	
6	Бурьгин Витя	1 мая 2003 г.	11 а		+7(921)004-34-34	
7	Петров Коля	4 июля 2006 г.	9 б		+7(911)004-14-14	
8	Сидоров Сеня	9 июля 2006 г.	9 б		+7(911)004-34-34	
9	Коровин Сева	19 октября 2006 г.	9 а		+7(921)009-18-19	
10	Мышкина Марина	3 февраля 2006 г.	10 б		+7(960)001-91-91	
11	Синичкина Лена	1 июня 2006 г.	9 б		+7(960)002-12-00	
12	Попова Ира	4 июля 2006 г.	9 а		+7(921)008-91-23	
13	Рысаков Рома	2 апреля 2004 г.	10 б		+7(911)007-65-43	
14	Крылов Дима	3 июня 2004 г.	11 а		+7(921)008-89-76	
15	Сидорова Оля	4 апреля 2004 г.	11 а		+7(911)009-87-65	
16	Потапов Федя	23 марта 2004 г.	11 б		+7(921)009-06-43	
17	Демина Света	11 мая 2004 г.	11 б		+7(960)008-76-44	
18	Карпова Саша	30 апреля 2004 г.	11 б		+7(921)003-56-46	
19	Перепелкина Оля	28 марта 2005 г.	10 б		+7(960)005-78-78	
20	Иванова Даша	23 мая 2005 г.	10 б		+7(921)004-46-75	
*	(No)					

Рис. 10. Список учеников

Теперь, когда мы закончили ввод данных, закроем таблицу, щелкнув по крестику в правом верхнем ее углу. Таким образом, с объектом Microsoft Access «Таблицы» мы можем работать в двух режимах: режиме таблицы (рабочий режим), когда вводим или редактируем записи, и режиме конструктора, когда создаем или редактируем структуру таблицы.

§ 1.3. Таблица «Преподаватели»

12 ⇔ 30 Сначала создадим вспомогательную таблицу-справочник, в которой будут храниться виды образования. Для этого, как и при создании предыдущей таблицы,

щелкнем по пункту верхнего меню «Создание», а затем по пиктограмме «Конструктор таблиц». В конструкторе определим два поля: «ID_образование» и «Образование». Первое поле типа «Счетчик» сделаем ключевым, второе – текстовой строкой из 15 символов (рис. 11а). Закроем

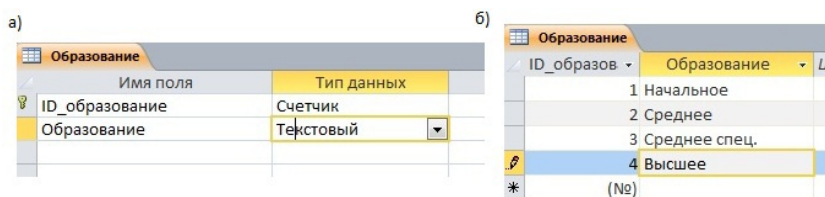


Рис. 11. Таблица «Образование»

конструктор с сохранением созданной структуры. Новой таблице дадим имя «Образование». В области «Все объекты Access» появилась новая таблица. Откроем таблицу двойным щелчком по ее имени и заполним, как показано на рис. 11б. Закроем таблицу. В ближайшее время она нам понадобится. А пока создадим в режиме конструктора таблицу, которая будет хранить данные о преподавателях. Определим ее поля, действуя, как в § 1.2:

1. ID_преподаватель. Счетчик. Это поле сразу можно сделать ключевым.

2. ФИО. Текстовая строка длиной в 50 символов. Поскольку без этого поля запись теряет смысл, сделаем это

поле обязательным. Остальные поля, как и прежде, оставим необязательными.

3. ДР. Определяется, как и в предыдущем параграфе.

4. Образование. Развернем меню в столбце «Тип данных» и выберем «Мастер подстановок». На экране появится первое диалоговое окно мастера (рис. 12). Установим

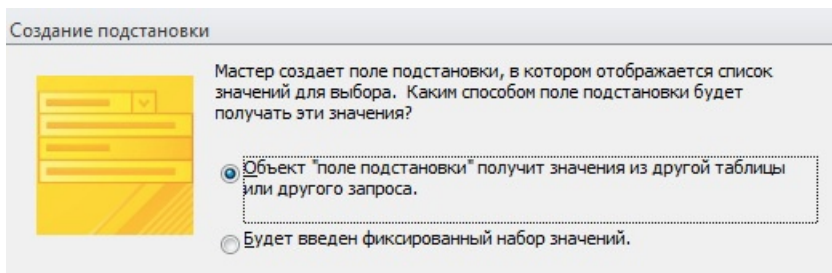


Рис. 12. Выбор способа подстановки

переключатель в положение «...получит значение из другой таблицы...» и нажмем «Далее». Появится окно, в котором нам предстоит сделать выбор таблицы (рис. 13а). Нас интересует таблица «Образование». «Далее». В следующем диалоговом окне перебросим интересующие нас поля из левой области «Доступные поля» в правую «Выбранные поля» (рис. 13б). Поскольку доступны всего два поля, содержащие код и соответствующий ему текст, нажмем двойную стрелку и перебросим сразу все. «Далее». Установим сортировку записей по возрастанию значений

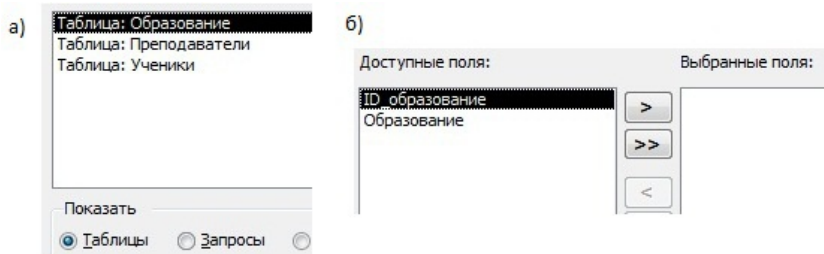


Рис. 13. Выбор таблицы и полей

поля «Образование» (рис. 14а), т. е. в алфавитном порядке. «Далее». См. рис. 14б. Опять «Далее». Оставим имя

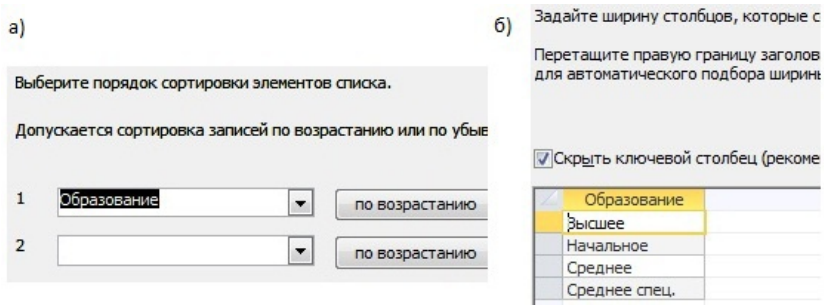


Рис. 14. Подстановка для поля «Образование»

поля «Образование». Поскольку мастер устанавливает связь с таблицей-справочником, система потребует сохранить таблицу (рис. 15). Сохраним таблицу под именем «Преподаватели». Обратите внимание, что после диалога мастер установил для поля тип данных «Числовой» (рис. 16).

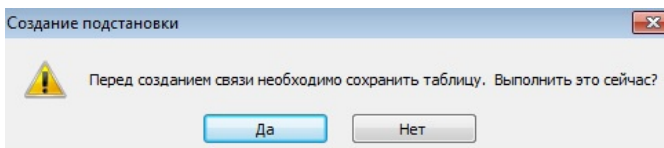


Рис. 15. Сохранение таблицы

Преподаватели	
Имя поля	Тип данных
ID_преподаватель	Счетчик
ФИО	Текстовый
ДР	Дата/время
Образование	Числовой
Должность	Текстовый
Телефон	Текстовый
Комментарии	Поле MEMO

Рис. 16. Структура таблицы «Преподаватели»

Пришло время пояснить, что же такое эта самая подстановка. Для этого рассмотрим более наглядный пример. Пусть мы работаем с базой данных большой фирмы. В базе есть таблица «Сотрудники», и в ней поле «Отдел» – текстовая строка с названием отдела, в котором работает сотрудник. Какие «приключения» ждут нас в этом случае? Во-первых, названия отделов бывают длинные и набор их отнял бы много времени у оператора. О нерациональном расходовании памяти мы уже не говорим. Во-вторых, люди не только могут делать ошибки, но

и делают их. Даже лишний пробел в названии отдела для системы означает другую текстовую строку. В базе стали бы появляться «двойники» отделов. В-третьих, если бы вдруг изменилось название одного из отделов, нам пришлось бы вручную менять его в каждой записи, где оно встречается. И если бы только в одной таблице. Поле «Отдел» может фигурировать во многих других таблицах. Уже страшно? Поэтому разумно во всех таблицах вместо названия хранить код отдела. Обычно код более компактен. Если изменится название отдела, код останется прежний. Соответствие между кодами и названиями отделов хранится в таблице-справочнике. Однако пользователь не должен держать в голове какие-либо коды. Тут и приходит на помощь «Мастер подстановок». В таблице хранится код, но мастер выводит на экран соответствующее значение из справочника. Здесь мы сталкиваемся с одним из проявлений принципа нормирования данных. В случае изменения названия отдела его потребуется изменить всего в одном поле одной записи. Несоблюдение этого принципа ведет к нарушению целостности данных.

5. Телефон. Создается так же, как и в предыдущем параграфе.

6. Должность. Текстовое поле, хранящее должность преподавателя по основному месту работы.

7. Комментарии. Поле МЕМО.

Закроем окно конструктора с сохранением и ту же таблицу откроем в рабочем режиме. Как вы помните, для этого достаточно сделать двойной щелчок по имени таблицы «Преподаватели». Заполнение полей больше не должно вызывать затруднений. Заметим, что значение поля «Образование» можно вводить не только из меню (рис. 17). Часто удобней начинать набор с клавиатуры. В данном

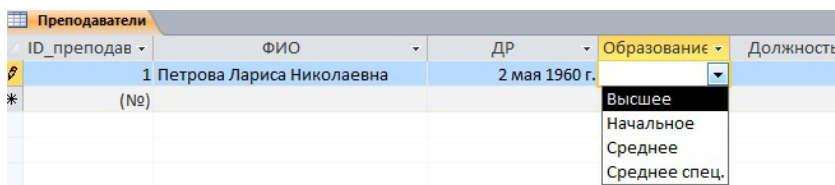


Рис. 17. Выбор значения поля в меню

случае, как только вы введете букву «в», установится значение поля «Высшее». Мы опять ввели небольшое количество записей (рис. 18). Ведь пока данные нужны только

ID_преп	ФИО	ДР	Образование	Должность	Телефон
1	Петрова Лариса Николаевна	2 мая 1960 г.	Высшее	Учитель математики	+7(921)002-12-00
2	Силин Иван Петрович	31 декабря 1965 г.	Высшее	Учитель физики	+7(921)003-13-11
3	Фролова Тамара Николаевна	26 апреля 1955 г.	Высшее	Учитель химии	+7(911)004-14-22
4	Солдатов Петр Иванович	12 августа 1980 г.	Среднее спец.	Учитель физкультуры	+7(911)004-34-33
5	Крылов Николай Кузьмич	18 октября 1970 г.	Среднее	Плотник	+7(960)009-76-44
6	Попова Светлана Борисовна	22 марта 1956 г.	Среднее	Пенсионерка	+7(921)003-56-55
(№)					

Рис. 18. Таблица «Преподаватели»

для тестирования приложения.

§ 1.4. Таблица «Кружки»

23 ⇔ 33 Отталкиваясь от свойств сущности «Кружки», создадим еще одну таблицу. Определим ее поля:

1. **ID_кружок.** Счетчик. Ключевое поле.
2. **Название.** Текстовое. Размер – 40. Обязательное.
3. **Преподаватель.** Начинаем диалог с «Мастером подстановок». Данные берем из таблицы «Преподаватели». Перебросим нужные нам поля – «ID_преподаватель» и «ФИО» – из левой области в правую (рис. 19), установим сортировку по возрастанию поля «ФИО».

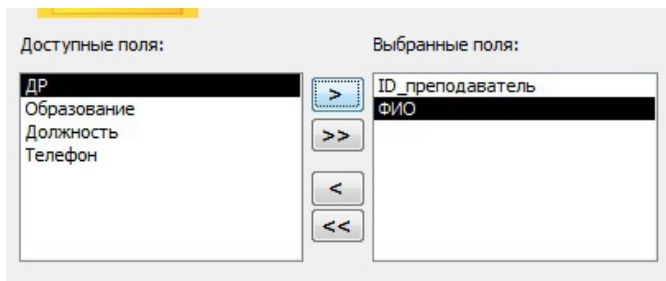


Рис. 19. Выбор полей для подстановки

Далее система покажет нам столбец подстановки (рис. 20). Имя поля оставим «Преподаватель». Поскольку мы установили связь с другой таблицей, система потребует сохранить текущую. Сохраним ее под именем «Кружки». Поле «Преподаватель» имеет тип данных «Числовой», т. е. на

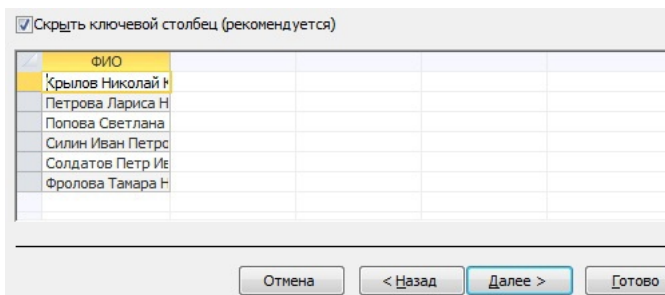


Рис. 20. Столбец подстановок

самом деле таблица хранит числовой код преподавателя.

4. Часы. Выберем тип данных «Числовой». В свойствах поля возьмем формат «Целое».

5. Комментарий. Поле МЕМО.

На рис. 21 показано окно конструктора. Закроем конструктор

Имя поля	Тип данных
ID_кружок	Счетчик
Название	Текстовый
Преподаватель	Числовой
Часы	Числовой
Комментарий	Поле МЕМО

Рис. 21. Структура таблицы «Кружки»

тор с сохранением результатов. Теперь в области «Все объекты Access» мы видим имена четырех таблиц (рис. 22). Откроем таблицу «Кружки» и введем несколько записей

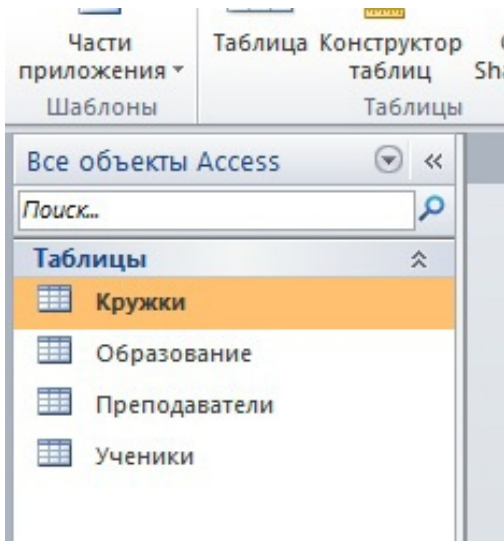


Рис. 22. Все объекты

(рис. 23). Конечно, кружков в школе должно быть

ID_кружок	Название	Преподаватель	Часы	Комм
1	Математический	Петрова Лариса Николае	68	
2	Физический	Силин Иван Петрович	40	
3	Моделирования	Силин Иван Петрович	60	
4	Баскетбольный	Солдатов Петр Иванович	80	
5	Химический	Фролова Тамара Никола	40	
6	Умелые руки	Крылов Николай Кузьми	35	
7	Вышивание	Попова Светлана Борисс	60	
*	(№)			

Рис. 23. Таблица «Кружки»

гораздо больше. Но нам сейчас много не надо.

Таким образом, мастер подстановок установил связь между таблицами «Кружки» и «Преподаватели». Связь между таблицами «Ученики» и «Кружки» пока не установлена. И мастер подстановок здесь нам не поможет.

§ 1.5. Связи

30 ⇔ 36 Таблицы «Ученики» и «Кружки» связаны более сложным образом. Такая связь может рассматриваться как еще одна сущность, и мы зададим ее при помощи таблицы. Эта таблица будет иметь три поля:

- 1. ID_ученик_кружок.** Счетчик. Ключевое поле.
- 2. Ученик.** В столбце «Тип данных» выбираем «Мастер подстановок». Подстановки берем из таблицы «Ученики». Выбранные поля – «ID_ученик» и «ФИ» (рис. 24). Сортировка по возрастанию значений «ФИ». Оставим имя поля «Ученик». При создании связи система потребует сохранить таблицу. Сохраним под именем «Ученик_кружок».
- 3. Кружок.** Снова используем «Мастер подстановок». Подстановки берем из таблицы «Кружки». Выберем поля «ID_кружок» и «Название» (рис. 25). Сортируем по возрастанию значений поля «Название».

Закроем конструктор с сохранением результатов и откроем таблицу для ввода данных. Пока заполним одну строку

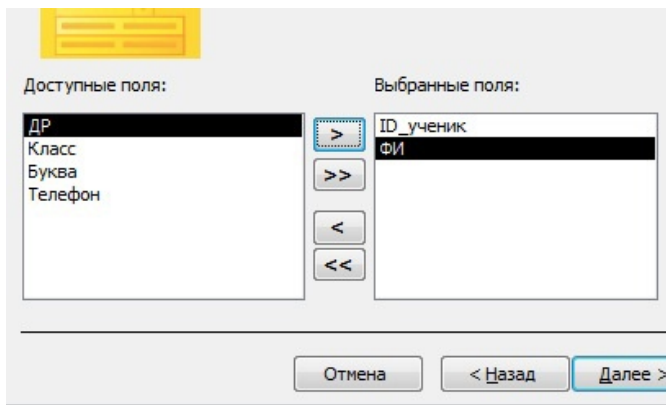


Рис. 24. Подстановка из таблицы «Ученики»



Рис. 25. Подстановка из таблицы «Кружки»

(рис. 26). Закроем таблицу. Теперь самое время рассмот-

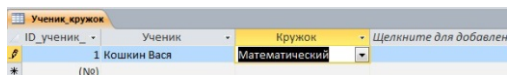


Рис. 26. Ввод данных в таблицу «Ученик_кружок»

реть систему связей между созданными нами таблицами. Выберем пункт меню «Работа с базами данных» и щелкнем по пиктограмме «Схема данных».

В открывшемся окне щелкнем правой клавишей мышки и выберем пункт меню «Отобразить все». В окне появятся все созданные нами таблицы и связи между ними. Здесь же можно установить новые связи. Для этого достаточно поместить указатель на интересующее нас поле одной таблицы, нажать левую клавишу и, не отпуская ее, установить указатель на соответствующем поле другой таблицы. После того как мы отпустим клавишу, появится линия, соединяющая соответствующие поля двух таблиц. Разумеется, не все поля можно и нужно связывать. Таблицы на схеме можно передвигать, чтобы добиться удобного для нас расположения. Разместим их так, как показано на рис. 27. На схеме видно, что наша база данных

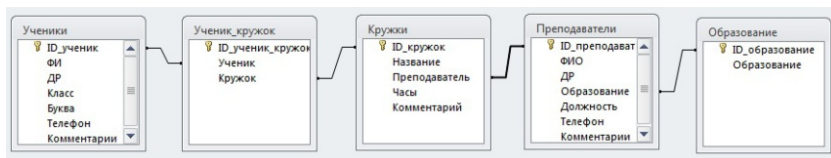


Рис. 27. Схема данных

представляет собой нечто целое, а не просто набор таблиц. Закроем окно схемы данных и займемся другим классом объектов.

§ 1.6. Запросы

33 ⇔ 43 В повседневной жизни под запросом понимают официальное обращение с требованием, просьбой о предоставлении каких-либо сведений. Если мы обращаемся к программной системе, просьба должна быть составлена на понятном ей языке. В нашем случае – это язык управления базами данных. К счастью, в Access почти все языковые проблемы решает «Конструктор запросов». Запросы бывают на выборку, модификацию и удаление данных, а также на создание таблиц. Но пока нас будут интересовать только запросы на выборку. Такие запросы позволяют отбирать записи по одному или нескольким признакам, собирать данные из разных таблиц, группировать записи и получать по группам обобщенную информацию: среднее, наибольшее или наименьшее значение поля, количество записей в группе. Возможности выбора в первую очередь определяются разнообразием полей. Результаты выполнения запросов система представляет в виде таблиц, внешне ничем не отличающихся от тех, которые мы только что создали. Поэтому иногда запрос на выборку называют логической таблицей. В этом параграфе мы создадим три запроса, содержащие все записи некоторых таблиц, но не все поля. Эти запросы пригодятся нам в дальнейшем.

1. Ученики Запрос. Этот запрос должен содержать список учеников, занимающихся в кружках. Выберем пункт меню «Создание» и щелкнем по пиктограмме «Мастер запросов». Мы могли бы создать запрос и непосредственно в режиме «Конструктор запросов». Однако мастер это делает быстрее. Достаточно лишь в процессе диалога объяснить мастеру, что мы хотим. В первом диалоговом окне выберем «Простой запрос», как показано на рис. 28. «Далее». Слева в верхней части следующего окна откроем

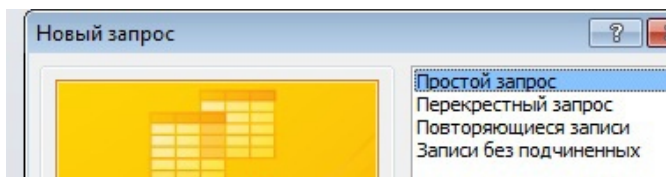


Рис. 28. Выбор вида запроса

меню и выберем источник данных для запроса – таблицу «Ученики». Пользуясь стрелочкой, перебросим слева направо интересующие нас поля: «ID_ученик», «ФИ», «Класс» и «Буква» (рис. 29). «Далее». Установим переключатель в положение «подробный» (рис. 30). «Далее». Мастер предлагает дать запросу имя «Ученики Запрос». Можно дать запросу другое имя, но мы оставим предложенное мастером. «Готово». На экране появится табличное представление запроса (рис. 31). Сразу можно заметить,

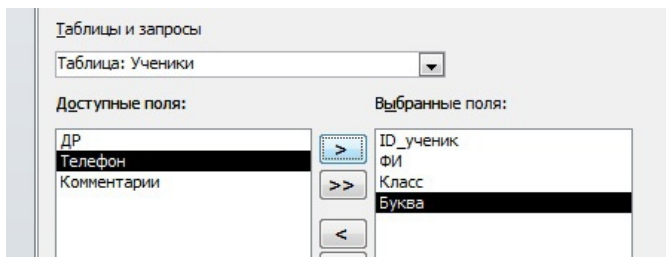


Рис. 29. Выбор таблицы и полей

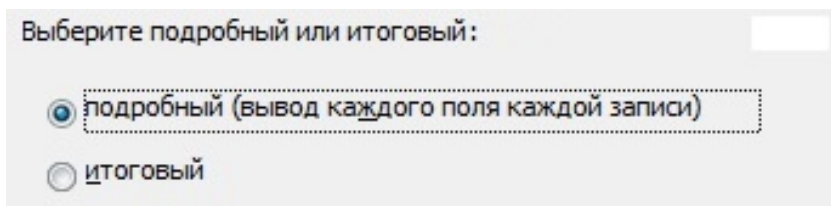


Рис. 30. Выбор подробного отчета

Все объекты Access		Ученики Запрос			
Поиск...		ID_ученик	ФИ	Класс	Буква
Таблицы	Кружки	1	Кошкин Вася	9 б	
	Образование	2	Тарасенко Гриша	10 а	
	Преподаватели	3	Лева Мухин	10 а	
	Ученик_кружок	4	Кузина Катя	10 б	
	Ученики	5	Бучиков Веня	9 а	
		6	Бурыйгин Витя	11 а	
		7	Петров Коля	9 б	
Запросы		8	Сидоров Сеня	9 б	
	Ученики Запрос	9	Коровин Сева	9 а	
		10	Мышкина Марина	10 б	
		11	Синичкина Лена	9 б	

Рис. 31. Табличное представление запроса

что искать ученика в таком списке неудобно. Хотелось бы видеть список в алфавитном порядке. Эту проблему мы сейчас решим. Обратим внимание, что в области «Все объекты Access» появился новый объект «Ученики Запрос». Закроем запрос. Щелкнем правой клавишей по имени запроса: «Ученики Запрос». В развернувшемся вертикальном меню выберем «Конструктор». На экране появится окно конструктора запросов (рис. 32). В верхней части окна

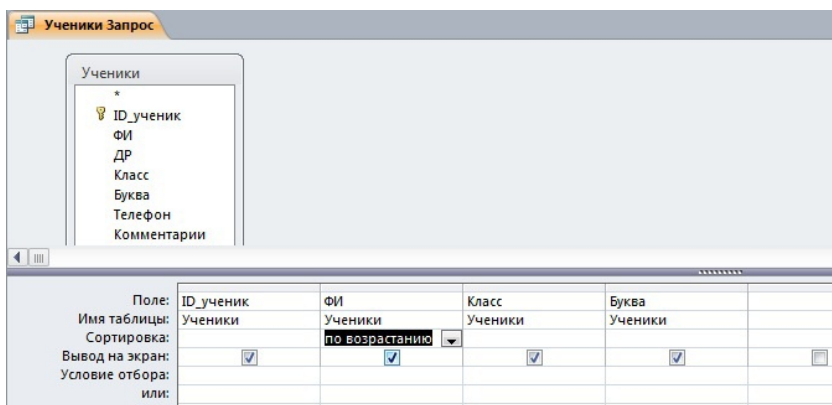


Рис. 32. Конструктор запросов

мы видим таблицу «Ученики», точнее список ее полей; внизу – выбранные поля. Каждому полю соответствует столбец таблицы. В первой строке даны имена полей, во второй – имя таблицы. Далее нам придется составлять запросы из данных разных таблиц и эта строка покажет,

из какой таблицы взято поле. Третья строка – «Сортировка». Откроем во втором столбце третьей строки меню и выберем «По возрастанию». Это значит, записи будут упорядочены в алфавитном порядке значений поля «ФИ». В четвертой строке изображены «птички». Если одну из «птичек» убрать, щелкнув по ней левой клавишей, соответствующее поле не будет выводиться на экран. В пятой строке записываются условия отбора данных. Это на случай, если мы захотим видеть не все, а только интересующие нас записи. Например, если в пятую строку столбца «Класс» ввести число 10, при открытии запроса в рабочем режиме мы увидим только записи об учениках 10-х классов. Но нас пока интересует общий список. Закроем конструктор, откроем запрос в рабочем режиме и убедимся, что записи действительно расположены по возрастанию поля «ФИ». Закроем запрос. Теперь нам предстоит создать еще два запроса. Диалог с мастером ведется, как и в случае первого запроса. Поэтому здесь воздержимся от подробных иллюстраций.

2. Преподаватели Запрос. Откроем «Мастер запросов». Выберем таблицу «Преподаватели», а в ней поля «ID_преподаватель» и «ФИО» (рис. 33). На рис. 34 показан запрос в рабочем режиме. Закроем запрос и откроем его в режиме конструктора. Зададим сортировку записей

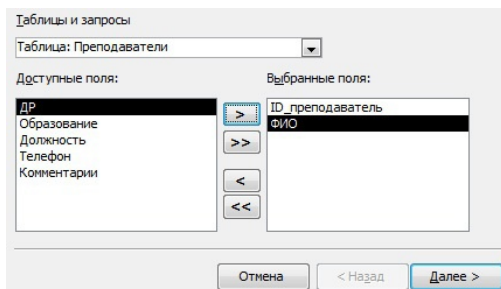


Рис. 33. Выбор таблицы и полей

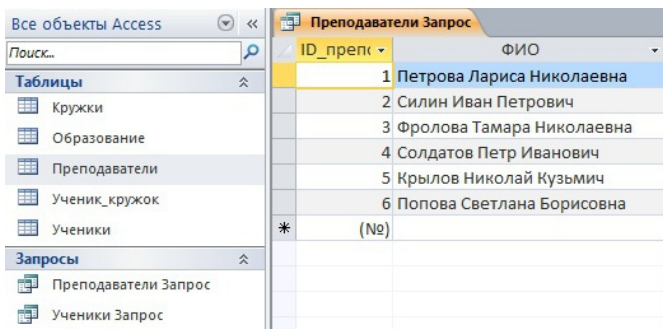
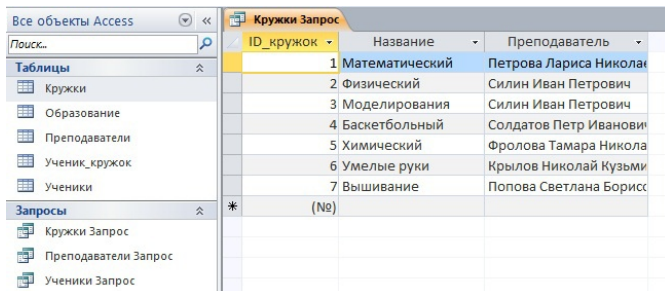


Рис. 34. Преподаватели Запрос

по возрастанию значений «ФИО». Закроем конструктор.

3. Кружки Запрос. Откроем «Мастер запросов». Выберем таблицу «Кружки», а в ней поля «ID_кружок», «Название» и «Преподаватели». Запрос сохраним под именем «Кружки Запрос». На рис. 35 показан запрос в рабочем режиме.



ID_кружок	Название	Преподаватель
1	Математический	Петрова Лариса Николаевна
2	Физический	Силин Иван Петрович
3	Моделирования	Силин Иван Петрович
4	Баскетбольный	Солдатов Петр Иванович
5	Химический	Фролова Тамара Николаевна
6	Умелые руки	Крылов Николай Кузьмич
7	Вышивание	Попова Светлана Борисовна
*	(№)	

Рис. 35. Схема данных

В дальнейшем нам часто придется создавать запросы на выборку как источники данных для различных объектов Microsoft Access.

Глава 2. Формы и макросы

Ввод, редактирование и просто просмотр данных непосредственно в таблицах и запросах очень неблагоприятное занятие. Во-первых, если таблица большая, много времени уходит на перемещение по строкам и столбцам. Во-вторых, некоторые поля, например МЕМО, изначально не предназначены для просмотра в таблице. В-третьих, иногда хочется видеть, по возможности целиком, только нужную запись. И наконец, для эффективной работы с базой данных требуются еще некоторые функции управления! Все эти проблемы решают формы.

§ 2.1. Форма «Ученики»

36 ⇔ 52 Начнем с создания формы, отображающей все поля записи в одном окне. Выберем пункт меню «Создание» и щелкнем по пиктограмме «Мастер форм». В первом диалоговом окне в качестве источника данных выберем таблицу «Ученики» и нажатием двойной стрелки перебросим все «Доступные поля» в список «Выбранные поля» (рис. 36). «Далее». Выберем вид формы «в один столбец» (рис. 37). «Далее». Мастер предлагает назвать форму, как и соответствующую таблицу, – «Ученики» (рис. 38).

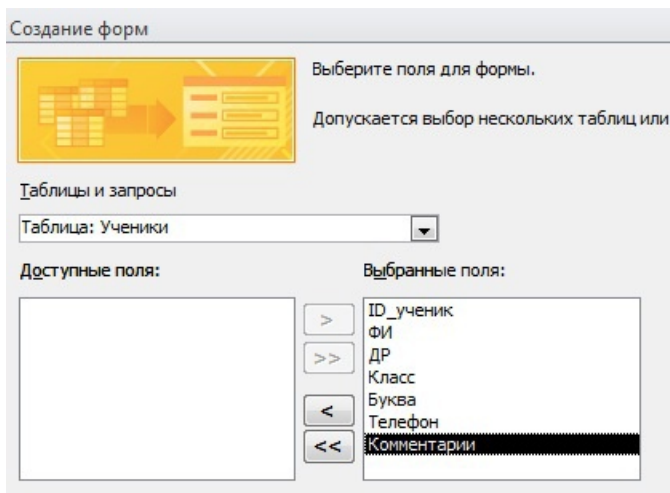


Рис. 36. Выбор таблицы и полей

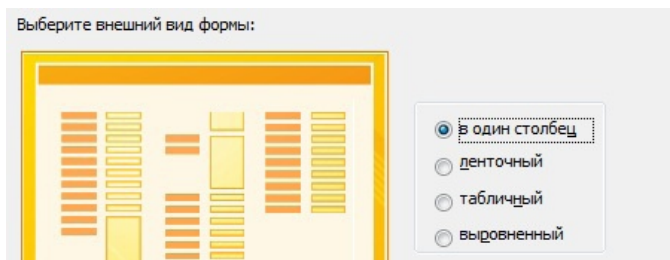


Рис. 37. Выбор внешнего вида формы

Оставим это название и нажмем «Готово». На экране откроется форма (рис. 39). В верхней части – «Заголовок формы», где в данный момент расположена надпись «Ученики»; ниже – «Область данных». Здесь представлены все поля записи с надписями. Если установить указатель

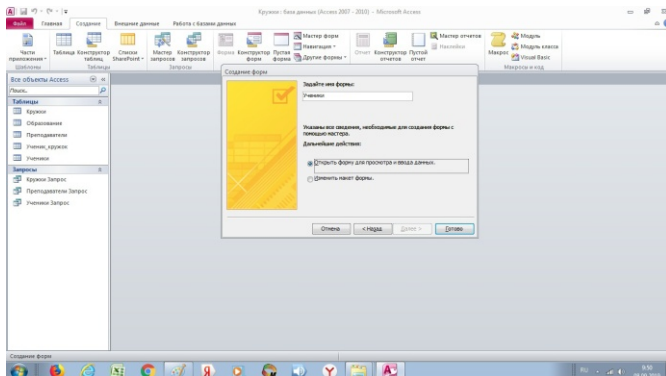


Рис. 38. Ввод имени формы

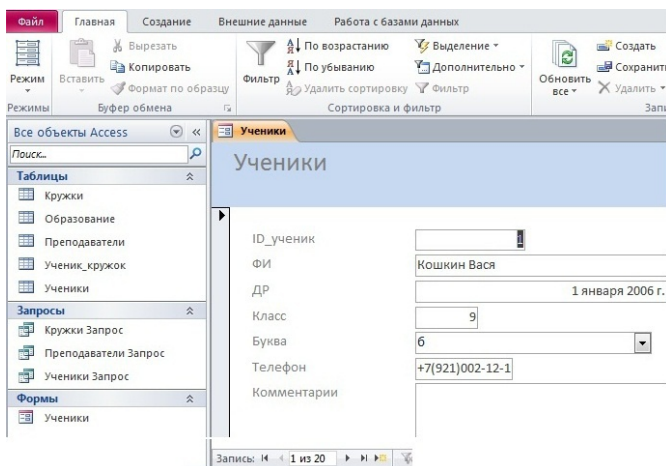


Рис. 39. Форма «Ученики»

в поле «Комментарии», то в левой его части появится полоса прокрутки. Слева – область выделения. После щелчка по ней левой клавишей область окрасится в черный цвет.

Если после этого нажать на клавиатуре «Delete», запись будет удалена. Но мы пока этого делать не будем! Внизу вы видите кнопки перехода. Сейчас у нас на экране отображена запись номер 1 из 20. С помощью стрелочек мы можем перемещаться по записям.

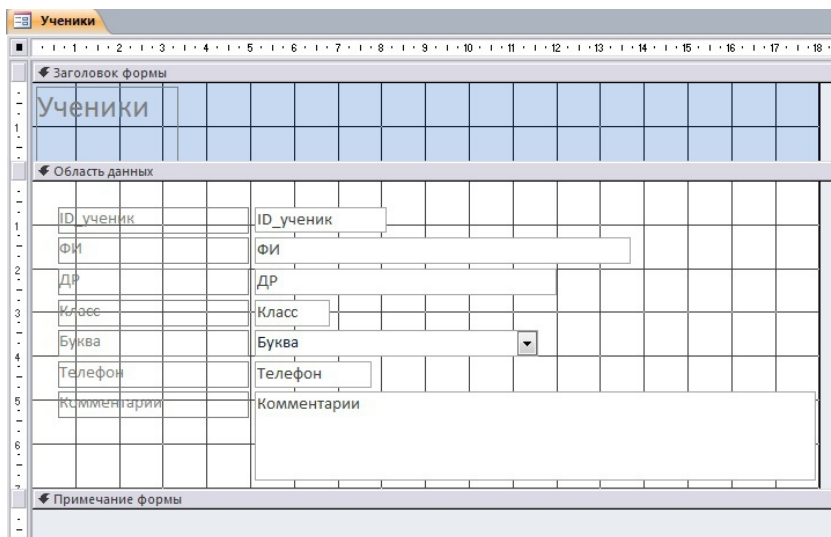


Рис. 40. Форма «Ученики» в режиме конструктора

В такой форме удобно работать с данными конкретного ученика, но трудно найти нужную запись. Теперь закроем форму и откроем в режиме конструктора (рис. 40). В этом режиме мы можем переставлять поля, менять надписи (сейчас перед каждым полем стоит надпись, совпадающая с его именем). Над каждой

областью расположена планка с названием этой области. В режиме конструктора мы обнаружим еще одну область, которую не видели в рабочем режиме, – «Примечание формы». Обратим внимание на набор значков в верхней части окна базы данных – «Панель элементов управления». В развернутом виде она выглядит, как на рис. 41.



Рис. 41. Панель элементов управления

Отсюда мы берем элементы управления для размещения в форме. Говорят, что форма является контейнером для элементов управления. Познакомимся с одним из них. Этот элемент называется «Кнопка» и на панели элементов управления обозначается прямоугольником с надписью «XXXX». Разместив в форме кнопку, мы можем приписать ей некоторые действия. В стандартных ситуациях с этой задачей прекрасно справляется мастер. Убедимся, что мастер подключен. Внизу панели элементов управления его

пиктограмма снабжена надписью «Использовать мастер», во включенном состоянии она окрашена в желтый цвет. Щелкнем по «XXXX», а потом где-нибудь в правой части примечания формы. Начнется диалог с мастером. Прежде всего выберем категорию действия и действие, которое должно произойти при нажатии кнопки. Как показано на рис. 42, мы выбрали категорию «Работа с формой» и действие «Закреть форму». «Далее». Выберем рисунок

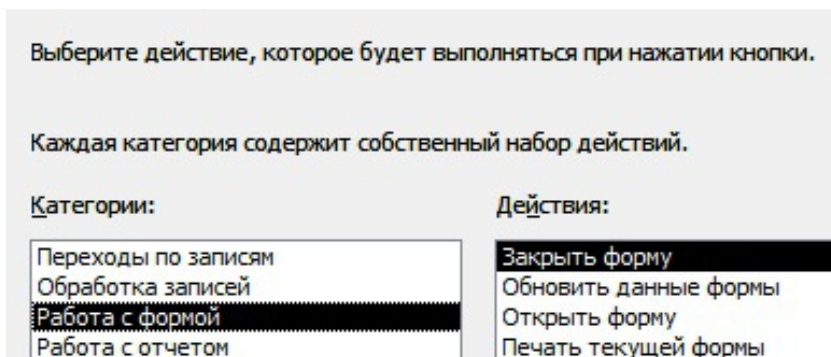


Рис. 42. Выбор категории и действия

с открытой для выхода дверью (рис. 43). «Далее». Дадим кнопке имя «Закреть» (рис. 44а) и нажмем «Готово». Заметим, что «Кнопка» – название элемента управления, а имя «Закреть» мы дали конкретной кнопке. В форме каждая кнопка и любой другой элемент управления должны иметь уникальное имя. И желательно, чтобы

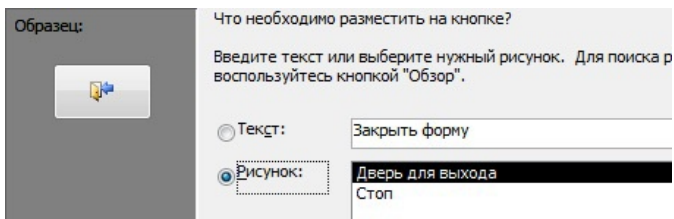


Рис. 43. Выбор рисунка

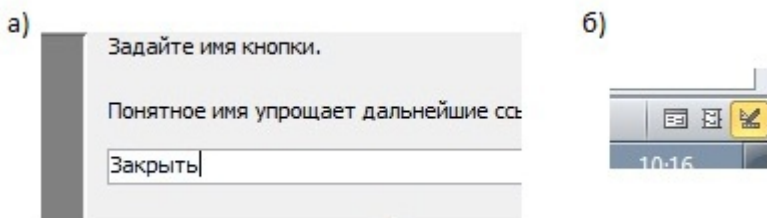


Рис. 44. а) Ввод имени кнопки; б) переключение режима работы

это имя отражало назначение элемента управления. Закроем конструктор с сохранением корректировок. Открыв форму в рабочем режиме, увидим, что в ней появилась кнопочка с дверью. Нажатие на эту кнопку приводит к закрытию формы.

Обратим внимание на пиктограммы в правом нижнем углу окна формы (рис. 44б). С их помощью мы можем оперативно менять режимы работы. В частности, нажатие на последнюю, на рисунке окрашенную в желтый цвет, переводит форму в режим конструктора. Нажатие на первую

возвращает форму в рабочий режим. Внимательно рассмотрим форму на предмет, что можно улучшить, и перейдем в режим конструктора. Удалим надпись «Ученики» из заголовка формы. Зацепим курсором планку с надписью «Область данных» и поднимем ее вверх. Код ID ученика не должен интересовать пользователя приложения. Мы его переместим в примечание формы. А надпись к нему просто удалим за ненадобностью. Подпись «ФИ» можно исправить на более понятную «Фамилия Имя». Подпись «ДР» – на «День рождения». Поле «Буква» вплотную придвинем к полю «Класс». Подпись удалим. В конечном итоге мы получили то, что видим на рис. 45. Закроем

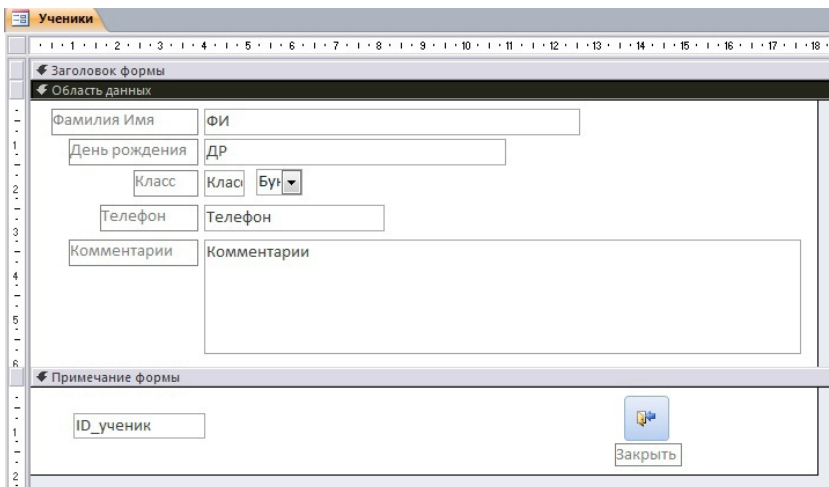


Рис. 45. Форма в режиме конструктора после редактирования

конструктор с сохранением корректировок. В рабочем режиме форма будет иметь вид, как на рис. 46.

Фамилия Имя: Кошкин Вася

День рождения: 1 января 2006 г.

Класс: 9 6

Телефон: +7(921)002-12-12

Комментарии:

Заккрыть

Рис. 46. Форма «Ученики» после редактирования

Общие	Подстановка
Формат поля	Длинный формат даты
Маска ввода	
Подпись	День рождения
Значение по умолчанию	

Рис. 47. Свойства поля «ДР»

В целом имена полей таблиц, созданных в первой главе, устраивают нас и как подписи. Но если мы хотим в дальнейшем избавиться от необходимости исправления подписи к полю «ДР», закроем форму и откроем таблицу «Ученики» в режиме конструктора. Установим указатель на поле «ДР». В свойствах поля установим «Подпись» –

«День рождения» (см. рис. 47). Теперь всюду будет фигурировать новая подпись, отличная от имени поля. Повторим проделанные выше манипуляции с таблицей «Преподаватели». К форме «Ученики» мы еще вернемся.

§ 2.2. Форма «Список учеников»

43 ⇔ 57 Следующую форму, как и предыдущую, создаем в процессе диалога с мастером. В качестве источника данных выберем запрос «Ученики Запрос» (рис. 48а).

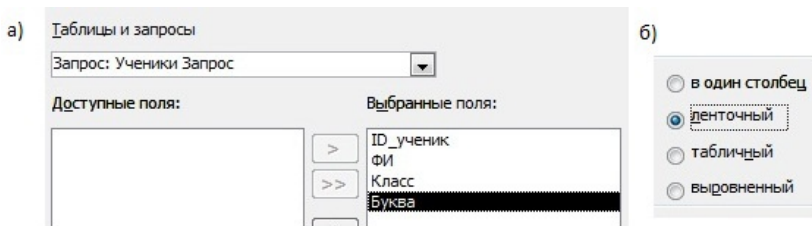
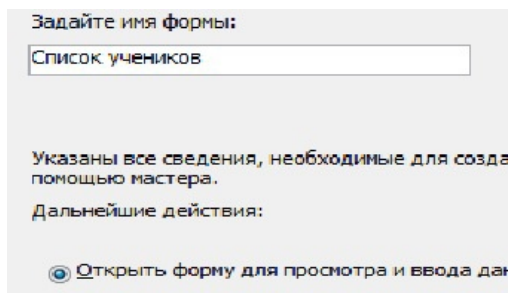


Рис. 48. Выбор запроса, полей и вида формы

Нажатием на двойную стрелку перебросим все поля в список выбранных. Источником данных для формы может быть как таблица, так и запрос. Внешний вид формы на этот раз выберем «ленточный» (рис. 48б). Дадим форме имя «Список учеников» (рис. 49). «Готово». На экране появится окно формы (рис. 50). Заметим:



Задайте имя формы:

Список учеников

Указаны все сведения, необходимые для создания формы с помощью мастера.

Дальнейшие действия:

Открыть форму для просмотра и ввода данных

Рис. 49. Ввод имени формы



ID_ученик	ФИ	Класс	Буква
6	Бурьгин Витя	11	а
5	Бучиков Веня	9	а
17	Демина Света	11	б
20	Иванова Даша	10	б

Рис. 50. Форма «Список учеников»

первой в списке стоит запись с номером 6. Это потому, что в источнике данных записи отсортированы по алфавиту. Структура окна в целом та же: сверху – заголовок формы, ниже – область данных, еще ниже – примечание формы. Слева от каждого поля – область выделения. Напомним, что если щелкнуть по этой области левой клавишей мыши, а затем нажать «Delete», то запись будет удалена. Перейдем в режим конструктора и внесем небольшие корректировки: из заголовка формы удалим надписи

«Список учеников», «ID_ученик» и «Буква». Сократим область заголовка. Ненужное пользователю поле «ID_ученик», расположенное в области данных, переместим в примечание формы. Откорректируем размеры полей и поле «Буква» придвинем поближе к полю «Класс». И наконец, создадим в примечании формы кнопку «Закрыть». Закроем конструктор с сохранением результатов и откроем форму в рабочем режиме (рис. 51). Убедимся

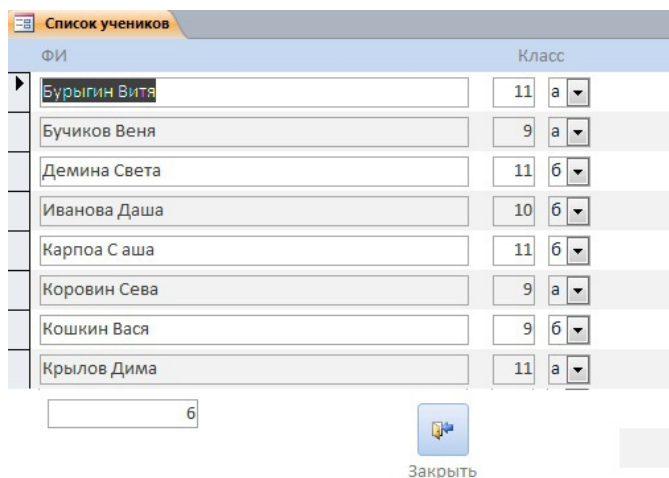


Рис. 51. Форма «Список учеников»

в том, что форма закрывается при нажатии на созданную нами кнопку.

Таким образом, мы создали две формы: одна позволит просмотреть упорядоченный список учеников и быстро

найти интересующего нас; другая предоставляет нам в удобном виде всю имеющуюся в базе информацию об ученике. Возникает вопрос: а нельзя ли их как-то объединить? Действительно, кнопочных дел мастер предоставляет нам такую возможность. Откроем форму «Список учеников» в режиме конструктора. Бросим в примечание формы кнопку и начнем диалог с мастером. Прежде всего сообщим ему, что кнопка будет выполнять работу с формой, а именно: она должна открыть форму (рис. 52а). Укажем, что кнопка должна открывать форму «Ученики»

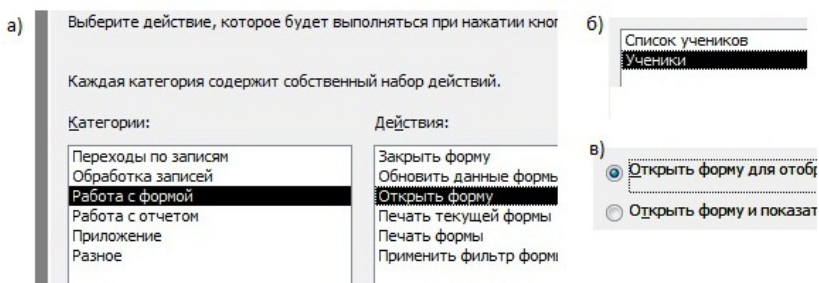


Рис. 52. Создание кнопки

(рис. 52б) и не для всех, а только отображенных записей (рис. 52в). Далее мы должны указать, какие поля используются для отбора записей. В диалоговом окне нам предлагают выбрать эти поля в формах «Список учеников» (слева) и «Ученики» (справа). В обеих формах выбираем поле «ID_ученик» (рис. 53). Теперь видно, что

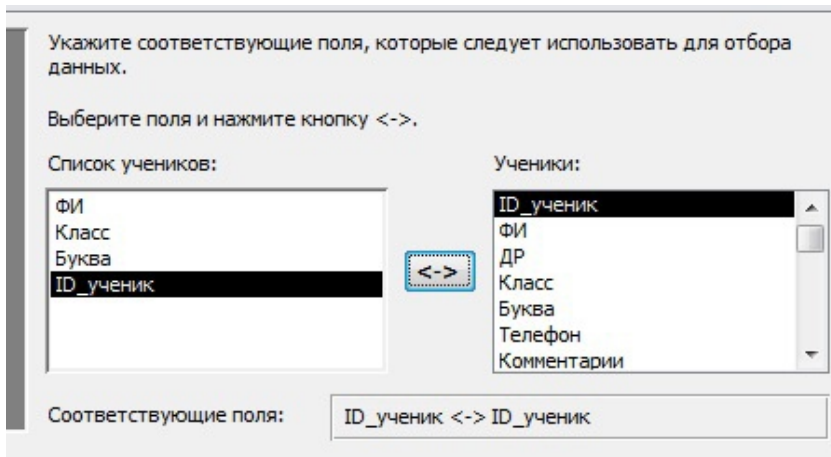


Рис. 53. Связь между формами

ненужное пользователю поле «ID_ученик» для разработчика очень даже полезно. После того как мы выделили эти поля, нажмем двойную стрелку посередине. Внизу диалогового окна за надписью «Соответствующие поля» появится текст: ID_ученик<->ID_ученик. Присвоим кнопке имя «Подробно», нажмем «Готово» и закроем конструктор. Откроем форму в рабочем режиме. Рядом с кнопкой «Закрыть» появилась кнопка «Подробно» (рис. 54). Убедимся, что при нажатии на эту кнопку открывается форма с подробной информацией о том ученике, на котором в списке мы установили указатель. У нас есть время подумать, что нам не нравится в только что созданных формах.

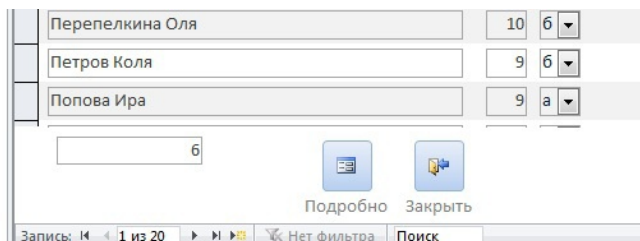


Рис. 54. Новая кнопка

А пока разберем один теоретический вопрос.

§ 2.3. Объекты, свойства, события и методы

52 ⇔ 69 Немного теории. Приложения Microsoft Access используют подход, за которым закрепилось название «объектно-ориентированное программирование» (ООП). Одно из ключевых понятий ООП – объект. Объекты делятся на классы. Каждый конкретный объект – представитель своего класса. В частности, объектом является форма. В свою очередь, форма – контейнер для элементов управления (надписи, поля, кнопки, переключатели, флажки и т. д.), которые также представляют собой объекты. Не следует отождествлять понятия «объект» и «класс объектов», хотя, например, под словом «кнопка» мы можем подразумевать как конкретную кнопку, так и кнопку вообще. Точно так же в быту словом «дерево»

мы иногда обозначаем дерево, растущее у нас под окном, а иногда класс растений. Класс объектов обладает некоторым набором присущих ему свойств. Каждый представитель класса имеет конкретные значения этих свойств. Например, класс «Поле» обладает свойством «Цвет фона» и поля одной формы, в общем случае, могут быть окрашены в разные цвета. С объектами могут происходить «события». Например, для формы – это «Открытие формы», «Закрытие формы» и т. д., для кнопки – «Нажатие кнопки», «Двойное нажатие кнопки» и т. д. Каждому событию может быть приписано действие, которое в ООП называют методом. Приведем известный каждому пользователю персонального компьютера пример. В окнах многих приложений Microsoft можно встретить пиктограмму с изображением ножниц. Если установить курсор на эту пиктограмму, всплывет подсказка «Вырезать». Хотя, на первый взгляд, ничего не случилось: у пиктограммы произошло событие «Получение фокуса», в ответ на это событие сработал «метод» – система выдала всплывающую подсказку. Откроем в режиме конструктора форму «Ученики». Ранее мы обратили внимание, что поле «ID_ученик» для пользователя не представляет интерес, но удалить его мы

не можем, так как оно нужно для установления связи между формами. Дабы оградить пользователя от ненужной ему информации, сделаем поле невидимым. Выделим поле «ID_ученик». В правой части окна базы данных откроется окно его свойств (рис. 55). Выберем вкладку

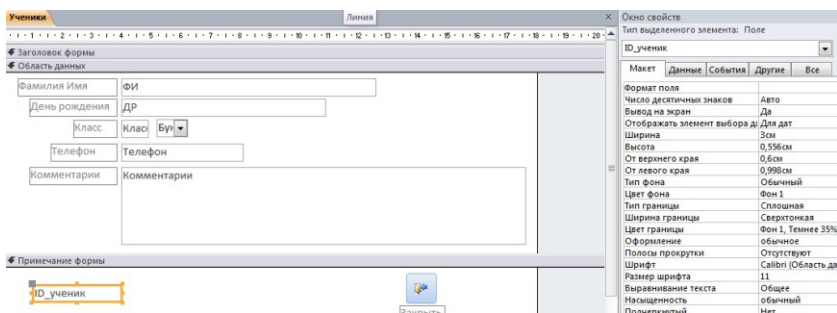


Рис. 55. Свойства поля «ID_ученик»

«Макет». Появится список свойств под заголовком «Формат поля». Найдем строку «Выход на экран». Мы видим, что этому свойству назначено значение «Да». Изменим его на «Нет» (рис. 56а), выбрав соответствующий пункт меню или введя текст вручную. Теперь в рабочем режиме это поле будет невидимым. Далее, если нам понадобится перейти к свойствам другого элемента управления, не закрывая окно свойств, достаточно сделать один щелчок левой клавишей по изображению соответствующего элемента. Чтобы увидеть свойства области данных (она тоже

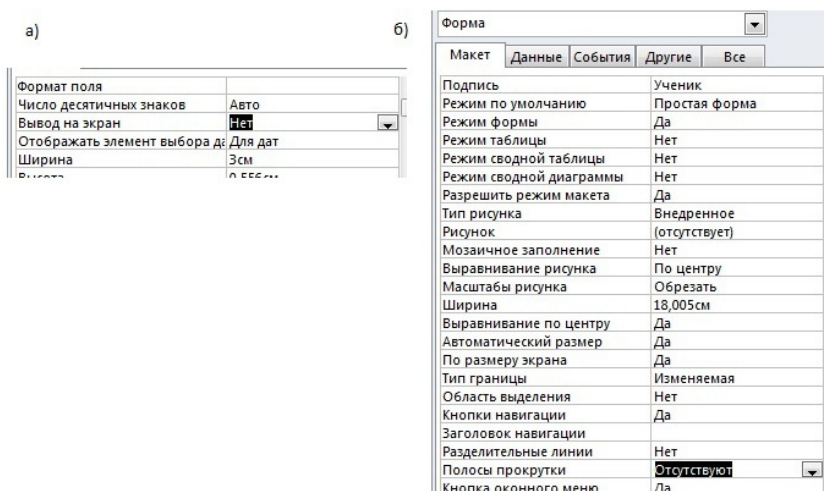


Рис. 56. Как сделать поле «невидимкой»

объект), надо найти в этой области данных пустое место, т. е. место, не занятое элементами управления, и щелкнуть по нему. А чтобы перейти к свойствам самой формы, надо щелкнуть по квадратику, расположенному в верхней части формы слева от линейки. Это мы сейчас и сделаем. На экране появятся свойства формы (рис. 56б). Изменим некоторые из них. Когда мы установили связь между формами «Ученики» и «Список учеников» (с. 55), то не подумали, какие конфликты из-за этого могут возникнуть. Например, мы удалим запись в форме «Ученики», а в форме «Список учеников» она останется. Конечно, система выдаст соответствующее сообщение, но лучше сразу

договориться, что удаление и добавление записей разрешено только в форме «Список учеников». В форме «Ученики» мы только просматриваем и редактируем записи. Значит, в ней не нужны область выделения и кнопки навигации. Отредактируем свойства на вкладке «Макет». В первой строке подпись «Ученики» можно исправить на «Ученик»; в строке «Область выделения» развернем меню и выберем пункт «Отсутствует»; «Кнопки навигации» – «Нет»; «Полосы прокрутки» – «Отсутствуют». Перейдем на вкладку «Данные». Установим значения: «Разрешить добавление» – «Нет»; «Разрешить удаление» – «Нет». На самом деле в процессе редактирования формы мы часто переходим в рабочий режим, чтобы посмотреть, что получилось, и опять возвращаемся в режим конструктора. Теперь мы умеем делать это быстро. На всякий случай напомним: для этих целей есть пиктограммы справа в нижней части окна формы (рис. 44 на с. 49). Что нам еще не нравится в созданной мастером форме? Пожалуй, она какая-то бледная, даже по сравнению с диалоговыми окнами мастера. Попробуем это исправить. Щелкнем по области данных. Справа появится окно ее свойств (рис. 57). В этом окне выберем строку «Цвет фона». Как только указатель окажется на этой строке, в правой ее части появится кнопка с текстом «...». Щелкнув по ней,

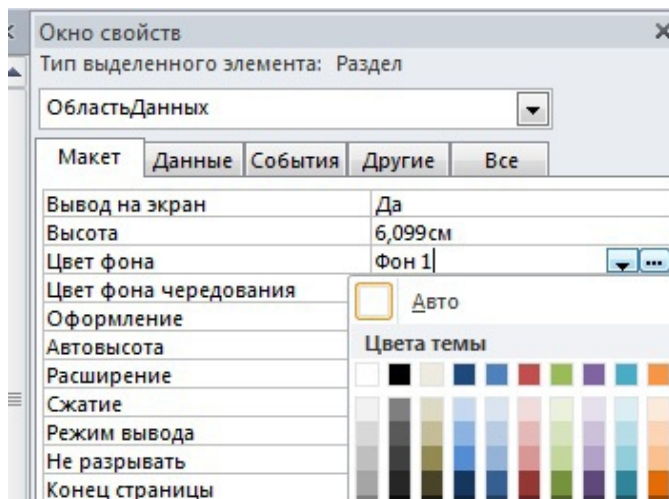


Рис. 57. Установка цвета фона

развернем набор цветов и выберем 5-й сверху в первой колонке. На новом фоне желательно изменить цвета текстов. Выделим все элементы управления, как показано на рис. 58. Для этого достаточно установить указатель

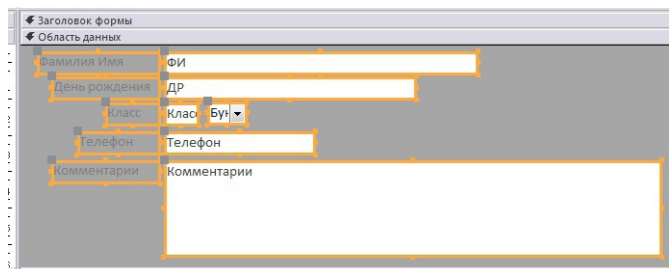


Рис. 58. Установка цвета шрифта

в одном из углов области данных, нажать левую клавишу мыши и, не отпуская ее, перевести указатель в противоположный угол. В окне свойств найдем строку «Цвет текста» и выберем черный цвет. Если теперь перейти в рабочий режим, форма будет иметь вид, как на рис. 59. Закроем




Рис. 59. Окончательный вид формы «Ученики»

форму «Ученики» и откроем в режиме конструктора форму «Список учеников». Перенесем в примечание и сделаем невидимым поле «ID_ученик». Установим цвета фона и текста. В списке часто для удобства просмотра цвета строк чередуются, т. е. четные и нечетные выделяются разными цветами. Если нам это не требуется, установим один цвет в строках «Цвет фона» и «Цвет чередования фона». Можно картинку на кнопке с надписью «Подробнее» заменить на «Бинокль». Для этого достаточно в свойствах кнопки установить указатель на строку «Рисунок», нажать «…»

и выбрать нужный рисунок. Заметим, что мы не учли еще одной ситуации. Если в форме «Список учеников» ввести фамилию и имя нового ученика и сразу же открыть форму «Ученики» нажатием кнопки «Подробнее», то ничего у нас не получится. Потому что новая запись еще не сохранена, т. е. в таблице «Ученики», которая является источником данных для одноименной формы, такой записи просто нет. Значит, предварительно мы должны сохранить новую запись. Бросим в примечание формы еще одну кнопку. В диалоговом окне укажем мастеру категорию действия «Обработка записей» и действие «Сохранить запись» (рис. 60). Дадим новой кнопке имя «Сохранить».

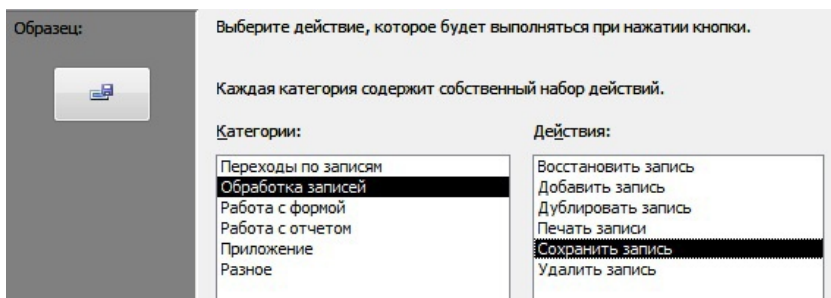


Рис. 60. Создание кнопки «Сохранить»

Поместим под кнопкой такую же надпись. Теперь в рабочем режиме форма «Список учеников» примет вид, как на рис. 61. Уже неплохо! Однако если пользователь будет



Рис. 61. Окончательный вид формы «Список учеников»

часто просматривать подробную информацию об учениках, ему неудобно будет каждый раз обращаться к расположенной внизу кнопке. Можно было бы привязать метод, т. е. некоторую процедуру, открывающую форму «Ученики», к событию «Нажатие кнопки». Но это событие происходит и тогда, когда мы хотим просто исправить ошибку в фамилии. Уточним, что под нажатием кнопки здесь подразумевается щелчок левой клавишей мыши. Щелчок не подходит. Остается двойной щелчок. Итак, при

двойным щелчке по полю «ФИ» должна открываться форма «Ученики». Здесь поле «ФИ» – объект, «Двойное нажатие» – событие, а методом будет «Внедренный макрос», который нам предстоит создать. Макрос – это последовательность макрокоманд. Макрокоманда, по сути, «большая команда». Макрокоманды могут выполнять такие стандартные действия, как открытие и закрытие форм и других объектов Access, переходы по записям и т. д. Внедренным называют макрос, привязанный к отдельно взятому объекту. Откроем форму «Список учеников» в режиме конструктора и рассмотрим окно свойств поля «ФИ» (рис. 62). В окне свойств поля выберем вкладку

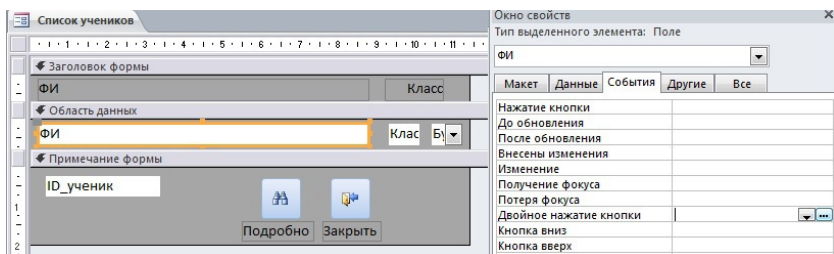


Рис. 62. Свойства поля «ФИ»

«События». Установим указатель на строке «Двойное нажатие кнопки». Щелкнем по появившейся кнопке «...». В правой части экрана появится «Каталог макрокоманд» (рис. 63). Определим для события «Двойное нажатие

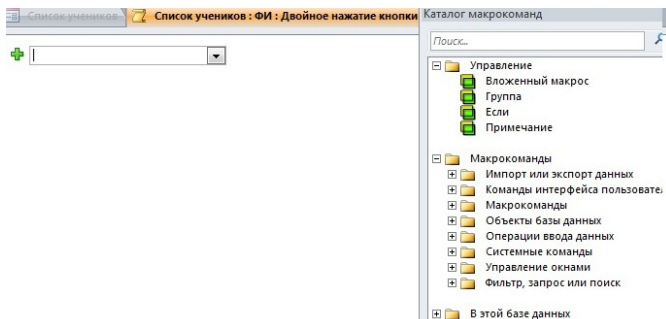


Рис. 63. Каталог макрокоманд

кнопки» метод. В меню выберем «Объекты базы данных», затем «Открыть форму». На экране появится окно параметров макрокоманды «Открыть форму» (рис. 64). Нам

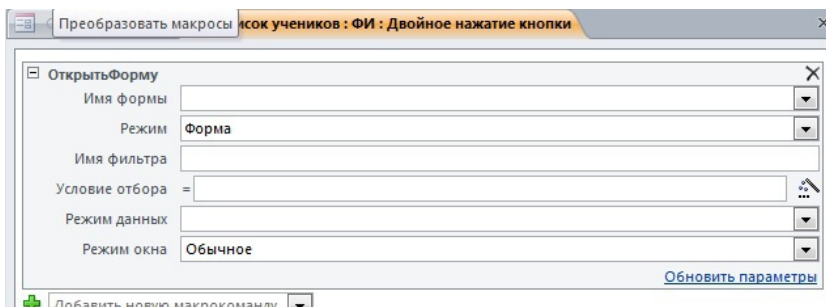


Рис. 64. Окно параметров макрокоманды «Открыть форму»

предстоит заполнить его. Развернем меню в строке «Имя формы» и выберем форму «Ученики» (рис. 65). В строке «Условия отбора» запишем:

[Формы]![Список учеников]![ID_ученик]=[ID_ученик].

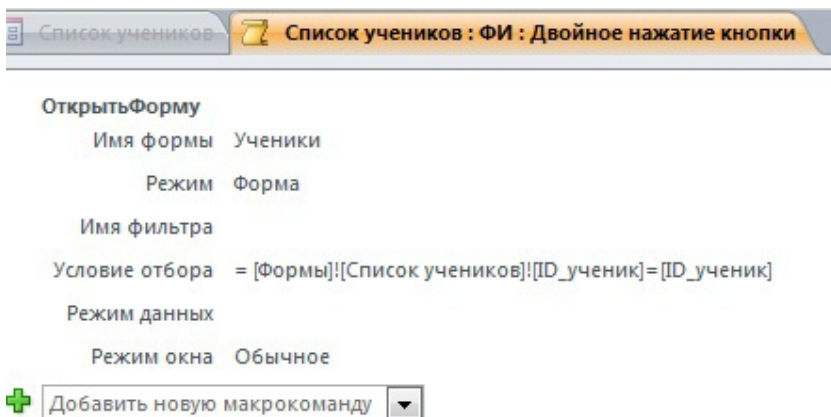


Рис. 65. Параметры макрокоманды

Здесь мы указали, что в форме «Ученики» будут просмотрены только те записи, у которых значение поля «ID_ученик» совпадает со значением аналогичного поля формы «Список учеников». Обратите внимание, что поле текущей формы задается только именем, а поле другой задается полностью: имя класса объектов, имя формы, имя поля. Все эти имена разделены символом «!». Квадратные скобки вокруг имен объектов при вводе выражения вручную можно не писать, если имена не содержат пробелов. Теперь закроем окно макросов и саму форму.

§ 2.4. Форма «Поиск учеников»

57 ⇔ 74 Даже если ученики расположены в алфавитном порядке, но когда список большой, искать в нем нужные фамилии, пользуясь полосой прокрутки, не всегда удобно. Особенно если фамилию помнишь неточно. На этот случай мы создадим еще одну форму. Поскольку новая форма не привязана ни к одному из источников данных, мы не будем прибегать к услугам мастера. Выберем пункт меню «Создание» и щелкнем левой клавишей по пиктограмме «Пустая форма» (рис. 66а). Появится пустая форма.

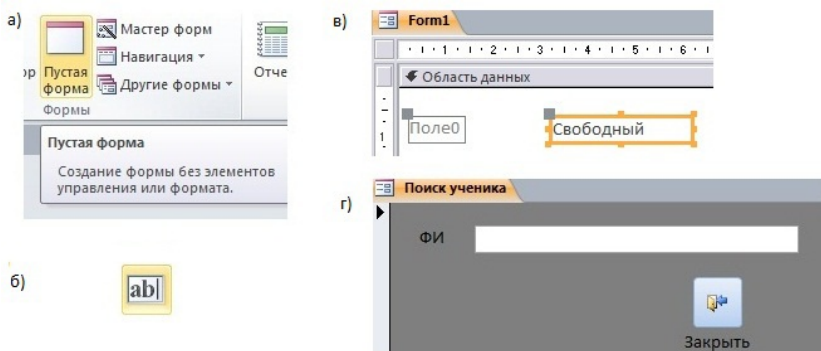


Рис. 66. Создание формы вручную

Перейдем в режим конструктора и с панели элементов управления возьмем «Поле» (рис. 66б). На экране появится «свободное» поле с надписью (рис. 66в). В это поле будем вводить символьную строку для поиска. Изменим надпись

на «ФИ». Растянем поле. Откроем свойства поля. Выберем вкладку «Другие» и дадим полю имя «ФИ». Перейдем на свойства формы, выберем вкладку «Макет» и введем подпись «Поиск ученика». Далее, как и в предыдущих формах, установим цвет фона в области данных. Разместим в области данных кнопку для закрытия формы. Кнопке дадим имя «Закрыть». С панели элементов возьмем объект «Надпись», разместим его под кнопкой и наберем в его области слово «Закрыть». Если теперь открыть форму в рабочем режиме, она будет иметь вид, как на рис. 66г. Закроем конструктор с сохранением результатов. Пока наша форма «ничего не умеет». Добавим еще одну кнопку (рис. 67). Бросим кнопку в область данных

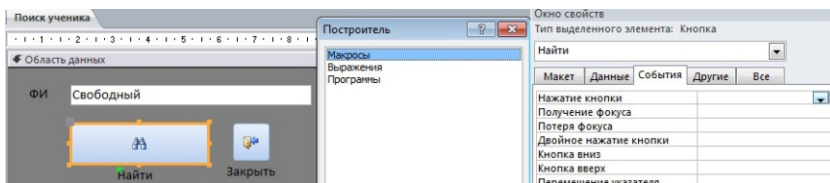


Рис. 67. Создание кнопки «Поиск»

рядом с кнопкой «Закрыть», но на этот раз уклонимся от диалога с мастером, нажав в первом же диалоговом окне кнопку «Отмена». Затем немного отредактируем внешний вид кнопки. Сделаем ее пошире, разместим под ней надпись «Поиск». Откроем свойства кнопки. Выберем для нее

рисунок «Бинокль». Рисунок принадлежит группе свойств «Макет». Перейдем на заставку «Другие». При создании кнопки система присвоила ей имя вроде «Кнопка7». Мы дадим ей имя, характеризующее ее назначение: «Найти». Теперь перейдем на заставку «События». На первом месте стоит самое важное событие, какое только может произойти с кнопкой, – «Нажатие кнопки». Поставим в соответствие этому событию метод. Установим указатель на соответствующей строке. Щелкнем по изображению «...». На экране появится окно «Построитель» (см. рис. 67). Выберем строку «Макросы». Откроется «Каталог макросов». Выберем «Объекты базы данных», а затем «Открыть форму». Теперь нам предстоит установить параметры макроса, которые указаны в окне с заголовком «Поиск ученика: Найти: Нажатие кнопки» (рис. 68). Заголовок говорит нам о том, что в форме «Поиск ученика» мы определяем действие для объекта «Найти», вызванное событием «Нажатие кнопки». Выбираем в меню форму, которую мы должны открыть, – «Список учеников». В строке «Условие отбора» вводим текст:

```
InStr(1;" " & [ФИ];Forms![Поиск ученика]![ФИ];1)>1.
```

Здесь мы впервые встречаемся с функцией VBA, т. е. Visual Basic for Applications. VBA – язык программирования, встроенный в Microsoft Access. В данный момент нас

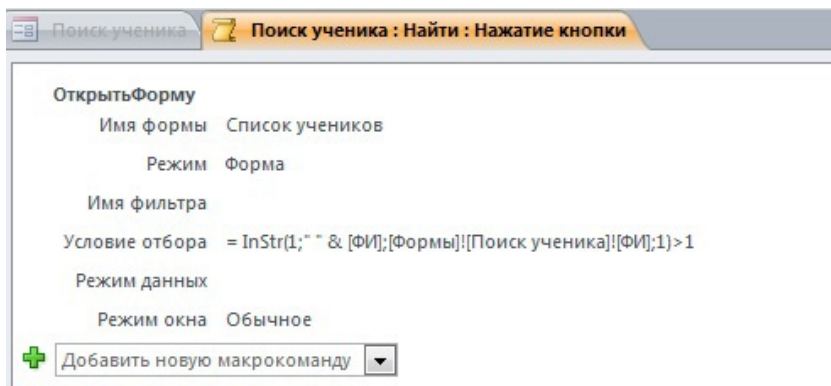


Рис. 68. Параметры макрокоманды

интересует только его функция InStr. Она определяет вхождение одной символьной строки в другую и имеет четыре аргумента. Первый задает позицию, с которой начинается поиск. Второй – строка, в которой осуществляется поиск (в данном случае это поле «ФИ» открываемой формы). Слева от поля [ФИ] пришлось добавить пробел, т. е. символьную строку, содержащую один пробел. Знак «&» здесь обозначает операцию сцепления строк. Вреда от этого пробела точно не будет, а прицепил его автор потому, что в новой версии функция почему-то перестала «замечать» первый символ. Третий аргумент – строка, вхождение которой в первую мы хотим установить, т. е. поле «ФИ» формы «Поиск ученика». Четвертый – тип сравнения (в нашем случае – 1). Функция возвращает позицию первого

вхождения одной строки в другую. Если это значение больше нуля, вхождение имеет место. Это и есть наше условие отбора. Возможно, читатель заметил, что слово «Формы» на этот раз автор написал по-английски. Это не нарочно, но система такую оплошность «проглотит». Также слово «Forms» не заключено в квадратные скобки. Их система потом добавит сама. Главное, чтобы названия объектов, содержащие пробел, заключались в скобки. Закроем конструктор с сохранением результатов и проверим, как работает поиск (рис. 69). Для этого откроем форму

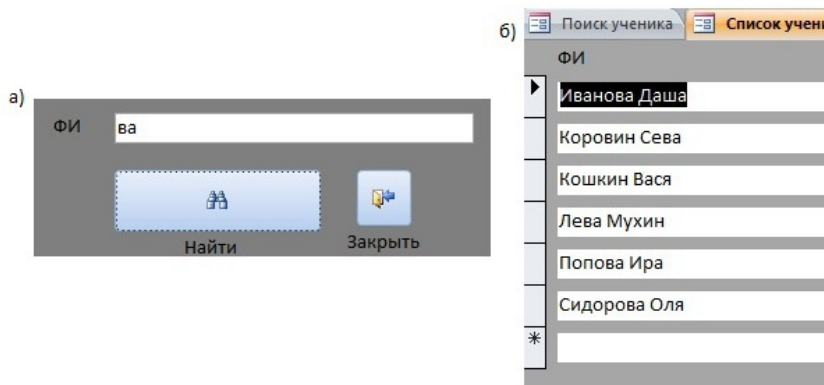


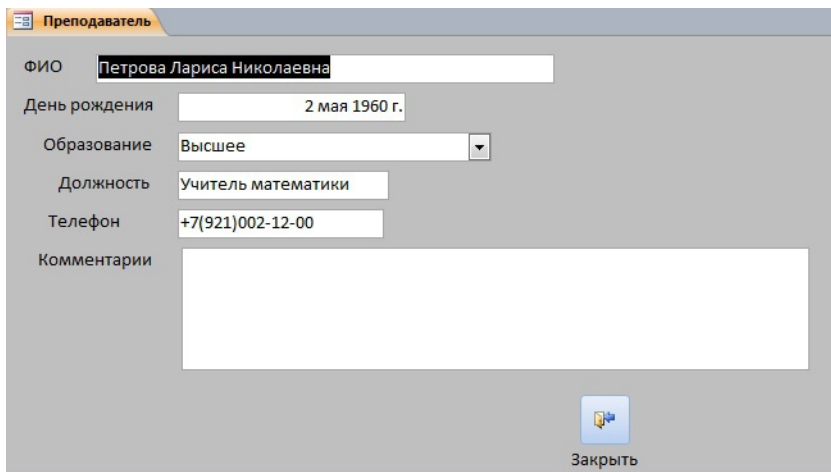
Рис. 69. Поиск ученика

«Поиск_ученика», в его поле «ФИ» введем «ва» и нажмем кнопку «Найти» (рис. 69а). В результате откроется форма

«Список учеников», содержащая список только тех учеников, в фамилии или имени которых содержится строка «ва» (см. рис. 69б). Разумеется, теперь вы можете, сделав двойной щелчок по полю с фамилией и именем, открыть форму с подробной информацией об ученике.

§ 2.5. Формы «для преподавателей»

69 ⇔ 76 До сих пор в этой главе мы занимались только формами для учеников. Вернемся к рис. 1 на с. 7 и обратимся к правой части рисунка. Нам предстоит создать три формы, аналогичные созданным ранее: «Преподаватели» (рис. 70), «Список преподавателей» (рис. 71) и «Поиск преподавателя» (рис. 72а). Но на этот раз читателю предоставляется возможность выполнить эту работу самостоятельно. Если вы тщательно проработали материал предыдущих параграфов, это не должно вызвать больших затруднений. В конце работы протестируйте созданный вами фрагмент приложения. Например, проверим работу формы «Поиск преподавателя». Для этого откроем форму и в поле «ФИО» введем «ло» (рис. 72а). После нажатия кнопки «Поиск» откроется форма «Список преподавателей» с набором всех записей, поле «ФИО» которых содержит подстроку «ло» (рис. 72б). Проверьте



Преподаватель

ФИО

День рождения

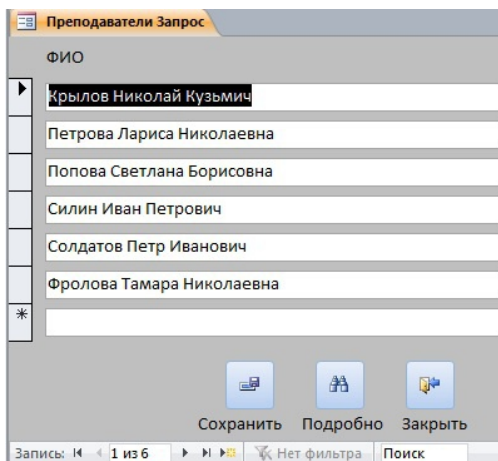
Образование

Должность

Телефон

Комментарии

Рис. 70. Форма «Преподаватели»



Преподаватели Запрос

ФИО

<input type="checkbox"/>	<input type="text" value="Крылов Николай Кузьмич"/>
<input type="checkbox"/>	<input type="text" value="Петрова Лариса Николаевна"/>
<input type="checkbox"/>	<input type="text" value="Попова Светлана Борисовна"/>
<input type="checkbox"/>	<input type="text" value="Силин Иван Петрович"/>
<input type="checkbox"/>	<input type="text" value="Солдатов Петр Иванович"/>
<input type="checkbox"/>	<input type="text" value="Фролова Тамара Николаевна"/>
<input type="checkbox"/>	<input type="text" value="*"/>

Записи:

Рис. 71. Форма «Список преподавателей»

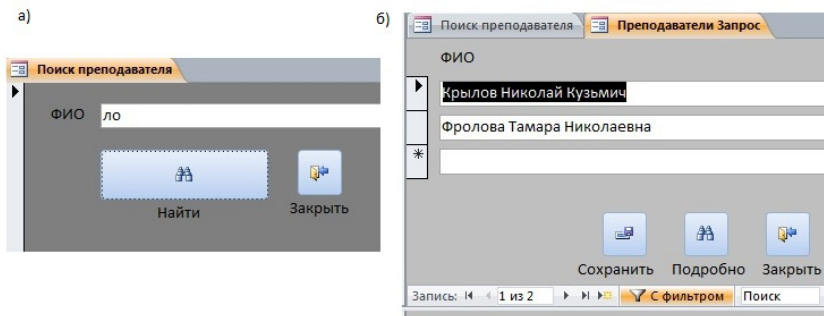


Рис. 72. Поиск преподавателя

также, открывается ли форма с подробной информацией о преподавателе при двойном щелчке по полю «ФИО».

§ 2.6. Формы «для кружков»

74 ⇔ 80 Снова обратимся к рис. 1 на с. 7. Разумеется, рисунок не есть концептуальная модель, но он напоминает нам, куда и зачем мы идем. Настало время заняться центральным понятием в нашей работе – кружками, т. е. тем, что изображено в центре рисунка. Точно так же, как и прежде, создадим «форму в столбец» (рис. 73), которая содержит подробную информацию о каждом кружке. Затем ленточную форму «Список кружков» (рис. 74). Разумеется, со всеми кнопками и всем тем набором возможностей, которые мы ранее устанавливали для

Кружок

Название

Преподаватель

Часы Комментарий

Рис. 73. Форма «Кружки»

Список кружков

Название	Преподаватель
<input type="text" value="Баскетбольный"/>	<input type="text" value="Солдатов Петр Иванович"/>
<input type="text" value="Вышивание"/>	<input type="text" value="Полова Светлана Борисовна"/>
<input type="text" value="Математический"/>	<input type="text" value="Петрова Лариса Николаевна"/>
<input type="text" value="Моделирования"/>	<input type="text" value="Силин Иван Петрович"/>
<input type="text" value="Умелые руки"/>	<input type="text" value="Крылов Николай Кузьмич"/>
<input type="text" value="Физический"/>	<input type="text" value="Силин Иван Петрович"/>
<input type="text" value="Химический"/>	<input type="text" value="Фролова Тамара Николаевна"/>
<input type="text" value="*"/>	<input type="text"/>

Запись: 1 из 7 | Нет фильтра | Поиск

Рис. 74. Форма «Список кружков»

аналогичных форм. Проверим формы в работе. На данный момент у нас имеется три «изолированных» фрагмента приложения. Вернемся к форме «Кружки». Теперь, когда мы освоили «внедренные макросы», т. е. макросы, приписанные к отдельным элементам управления, у нас может появиться желание сделать так, чтобы из формы с подробной информацией о кружке можно было по двойному щелчку по полю «Преподаватель» получить информацию о преподавателе. Для этого в режиме конструктора в форме «Кружки» откроем окно свойств поля «Преподаватель». Выберем вкладку «События» и поставим в соответствие событию «Двойное нажатие кнопки» внедренный макрос. Для этого щелкнем по кнопке « \cdots », в окне «Каталог макросов» выберем «Объекты базы данных», а затем «Открыть форму». На экране появится окно параметров макрокоманды (рис. 75). Заполним окно, как показано на рис. 75. В строке «Имя формы» укажем «Преподаватели», т. е. зададим имя той формы, которую мы хотим открыть. В строке «Условия отбора» задаем условия, по которым отбираются записи из указанной формы:

[ID_преподаватель]=[Формы]![Кружки]![Преподаватель].

Напомним, что в поле «Преподаватель» формы «Кружки» на самом деле хранится код преподавателя, но мастер подстановок вместо кода показывает соответствующий ему

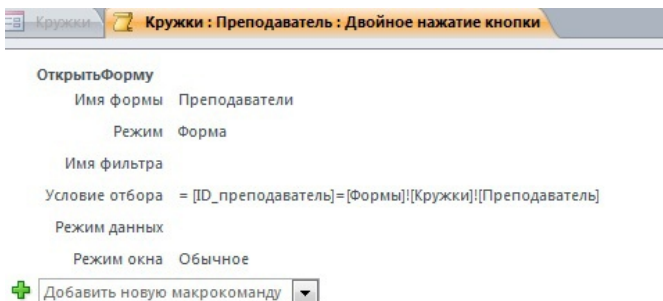


Рис. 75. Параметры макрокоманды

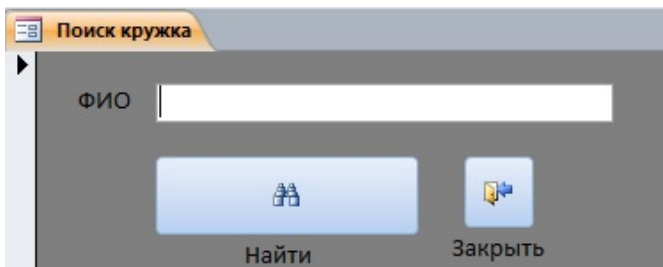


Рис. 76. Поиск кружка

текст. Осталось закрыть конструктор с сохранением результатов корректировок. Откроем форму в рабочем режиме и проверим ее работу. По аналогии с формами «Поиск ученика» и «Поиск преподавателя» создадим форму «Поиск кружка» (см. рис. 76). Заметим, что пока в наших формах никоим образом не отражена связь между сущностями «Кружки» и «Ученики». Этой связью нам и предстоит заняться в дальнейшем.

§ 2.7. Подчиненные формы

76 ⇔ 89 Ранее мы заметили, что связь между таблицами «Кружки» и «Ученики» имеет более сложный характер, чем между таблицами «Кружки» и «Преподаватели». Для задания этой связи мы в § 1.5 на с. 30 создали таблицу «Ученик_кружок». Теперь в процессе диалога с мастером построим соответствующую форму. Выберем таблицу «Ученик_кружок» и только одно ее поле «Ученик» (рис. 77). «Далее». Вид формы «ленточный» (рис. 78).

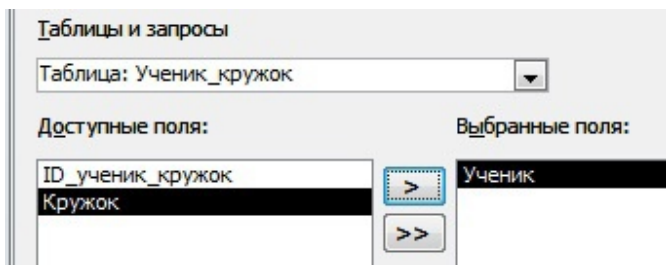


Рис. 77. Выбор таблицы и полей

«Далее». Сохраним форму под тем же именем, что и соответствующую таблицу (рис. 79). «Готово». На экране появится новая форма (рис. 80). В ней пока мы видим только одну запись, поскольку только одну и заполнили (рис. 26 на с. 34). Отредактируем форму «Ученик_кружок» в режиме конструктора (рис. 81а). Добавим в ее примечание только одну кнопку «Сохранить», которая отвечает

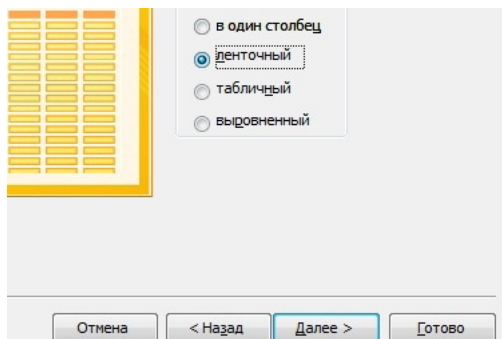


Рис. 78. Выбор типа формы

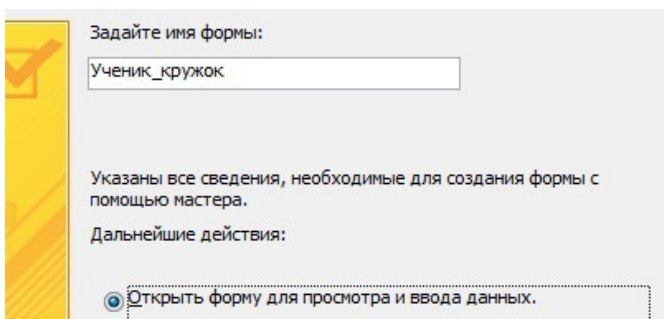


Рис. 79. Ввод имени формы

за сохранение новой записи. В рабочем режиме форма будет иметь вид, как на рис. 81б. Форма имеет единственное поле «Ученик». У читателя может возникнуть вопрос: нельзя ли снова сделать так, чтобы по двойному щелчку по этому полю открывалась форма с подробной информацией об ученике? Попробуем. В окне свойств поля «Ученик» выберем вкладку «События» (рис. 82). Установим

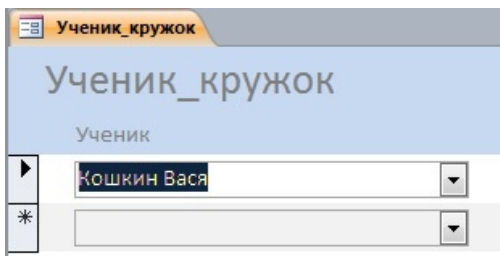


Рис. 80. Форма «Ученик_кружок»

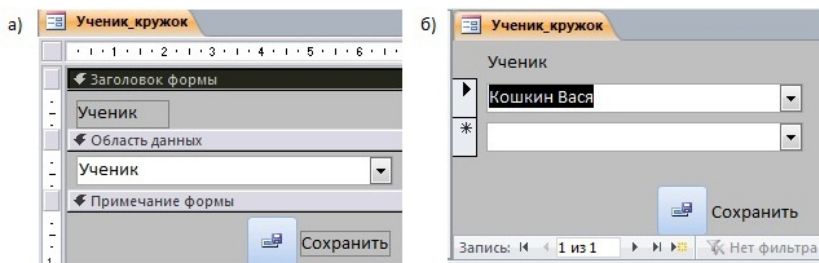


Рис. 81. Форма «Ученик_кружок» после редактирования

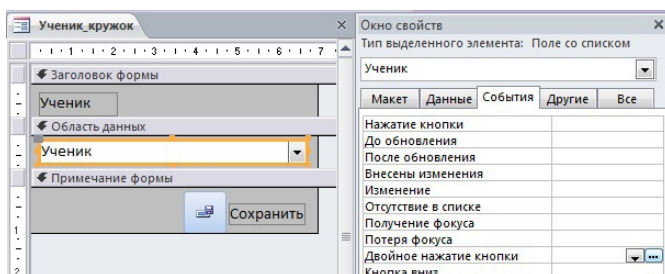


Рис. 82. Свойства поля «Ученик»

указатель на строке «Двойное нажатие кнопки». Щелкнем по значку «...». Откроется «Построитель» (рис. 83).

Выберем строку «Макросы». Мы хотим реализовать

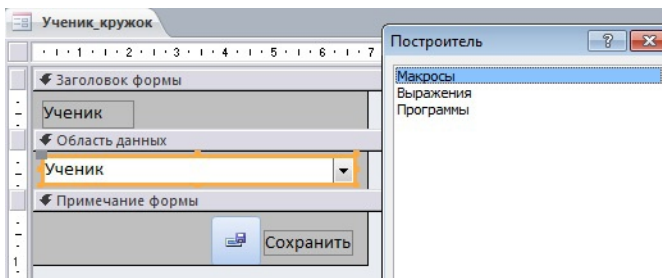


Рис. 83. «Построитель»

метод, связанный с событием «Двойное нажатие кнопки». В окне «Каталог макрокоманд» выберем «Объекты базы данных», а затем «Открыть форму» (рис. 84).

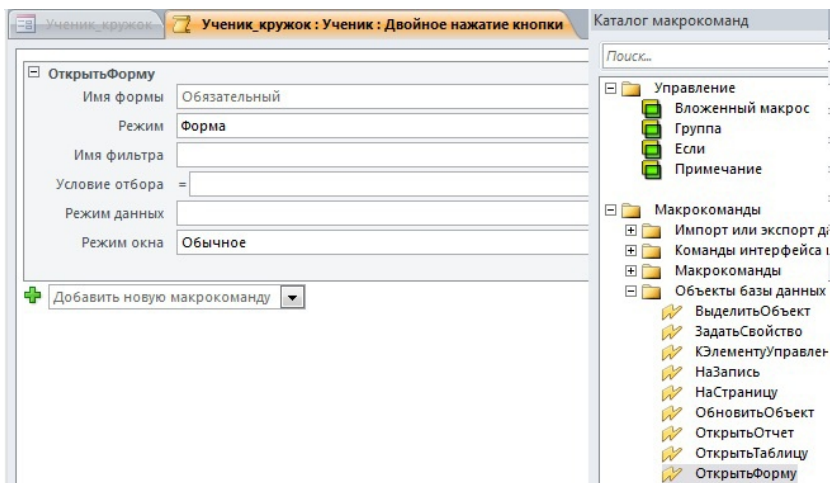
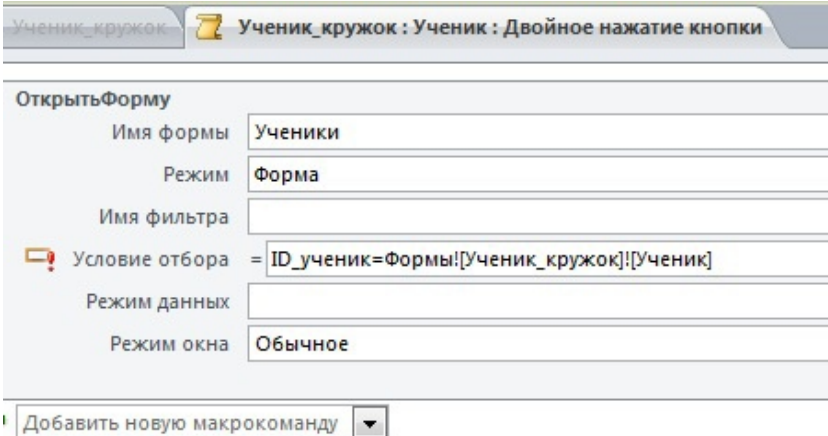


Рис. 84. Каталог макрокоманд

Зададим параметры макрокоманды «Открыть форму». Прежде всего, это имя формы, которую мы хотим открыть: «Ученики» (рис. 85). «Условие отбора»:



ОткрытьФорму	
Имя формы	Ученики
Режим	Форма
Имя фильтра	
Условие отбора	= ID_ученик=Формы![Ученик_кружок]![Ученик]
Режим данных	
Режим окна	Обычное

Добавить новую макрокоманду ▼

Рис. 85. Параметры макрокоманды

ID_ученик=Формы![Ученик_кружок]![Ученик].

Закроем конструктор с сохранением результатов. Откроем форму в рабочем режиме и убедимся, что двойной щелчок по полю «Ученик» вызывает форму «Ученики». Откроем в режиме конструктора форму «Кружки». Зацепим указателем нижний правый ее угол и расширим прямоугольник формы. В свободное пространство в правой части формы мы собираемся поместить элемент управления, который называется «Подчиненная форма». Возьмем его с панели элементов (рис. 86) так же, как мы брали кнопки

и т. д. На экране появится первое диалоговое окно (рис. 87).



Рис. 86. Панель элементов

В этом окне выберем форму, которую хотим разместить

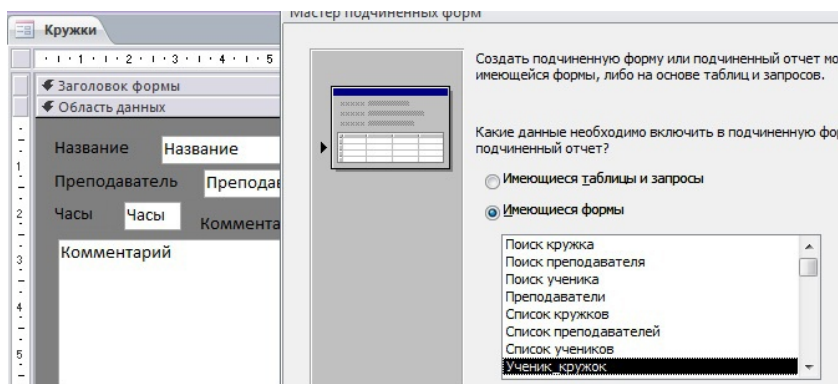


Рис. 87. Выбор формы

в области данных формы «Кружки». Это форма «Ученик_кружок». В следующем окне выберем поле связи «ID_ученик» (рис. 88). Осталось дать новому элементу управления имя «Ученик_кружок». Это имя совпадает с именем известной нам формы. Посмотрим, что

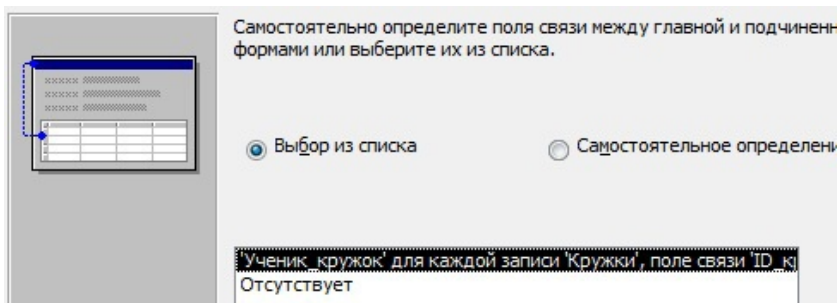


Рис. 88. Выбор поля связи

получилось. В области формы «Кружки» появилась форма «Ученик_кружок» (рис. 89). Не следует думать, что

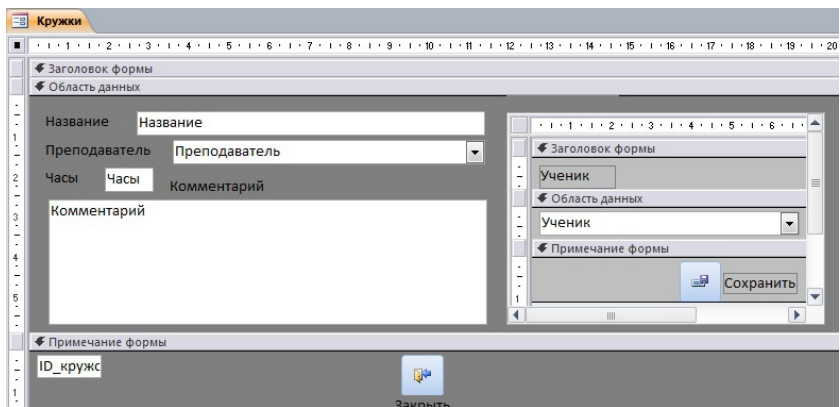


Рис. 89. Подчиненная форма в режиме конструктора

элемент управления и одноименная подчиненная форма суть одно и то же. Элемент управления здесь выступает в качестве механизма, который обеспечивает связь форм.

Закроем конструктор с сохранением результатов и протестируем приложение. Откроем форму «Список кружков». Дважды щелкнем в этом списке по названию кружка «Вышивание». Откроется форма «Кружки» (рис. 90). Вроде

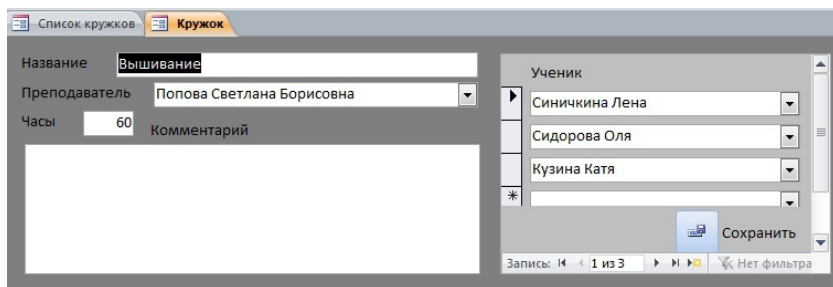


Рис. 90. Подчиненная форма в рабочем режиме

все неплохо. Как и прежде, если сделать двойной щелчок по полю «Преподаватель», откроется форма с подробной информацией о преподавателе. Но вот беда: если теперь мы сделаем двойной щелчок по полю с фамилией и именем интересующего нас ученика, на экране появится диалоговое окно с просьбой ввести его код. Система не определила этот код. Дело в том, что, когда форма становится подчиненной, она частично теряет свой «суверенитет». Теперь нам понадобится внести исправления в «Условие отбора». Чтобы обратиться к полю «Ученик», мы должны сослаться на «вышестоящую инстанцию». Откроем в режиме конструктора окно «Кружки», выйдем на свойства

поля «Ученик» и т. д., пока не увидим окно параметров макрокоманды (рис. 91). Внесем изменения в строку

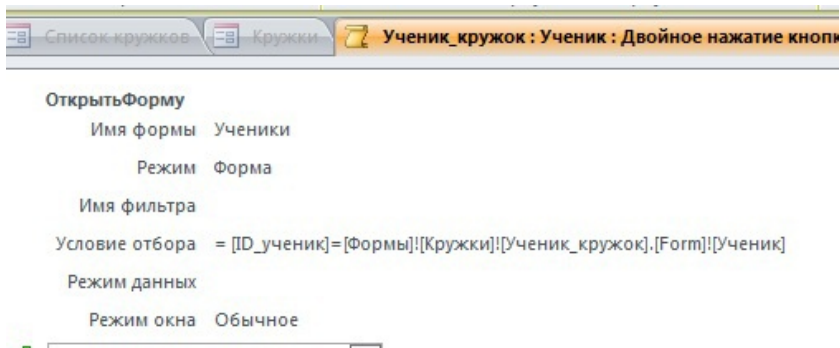


Рис. 91. Параметры макрокоманды

«Условия отбора». Теперь строка должна выглядеть так: `[ID_ученик]=[Формы]![Кружки]![Ученик_кружок].[Form]![Ученик]`. После знака равенства сначала мы ссылаемся на элемент управления: `[Формы]![Кружки]![Ученик_кружок]`. Затем, после «точки», указываем интересное нас поле в этом элементе управления. Следует запомнить, что здесь следует писать не `[Форма]`, а `[Form]`. Не потому, что в этом кроется какой-то глубокий смысл, а потому, что иначе не будет работать.

Глава 3. Кнопочная форма

§ 3.1. Диспетчер кнопочных форм

80 ⇔ 93 Далее мы могли бы заняться созданием других фрагментов приложения, но не будем злоупотреблять терпением читателя. Он имеет право наконец увидеть нечто целое, объединяющее все созданные ранее объекты базы данных. Для этой цели задействуем «Диспетчер кнопочных форм». В окне Access 10 вы можете не найти соответствующей пиктограммы, поскольку в этой версии имеются и другие средства навигации. Некоторые думают, что этого диспетчера здесь и вовсе нет. Однако пиктограмму «Диспетчер кнопочных форм» надо просто предварительно установить на панели быстрого доступа. Для этого выберем пункт верхнего горизонтального меню «Файл». В развернувшемся вертикальном меню щелкнем по строке «Параметры». На экране появится окно «Параметры Access» (рис. 92). В меню его левой части выберем строку «Панель быстрого доступа». На экране появится окно с заголовком «Настройка панели быстрого доступа». Ниже – список команд. Этот список представлен несколькими наборами. По умолчанию установлены «Часто используемые команды» (рис. 93). Развернем меню и выберем

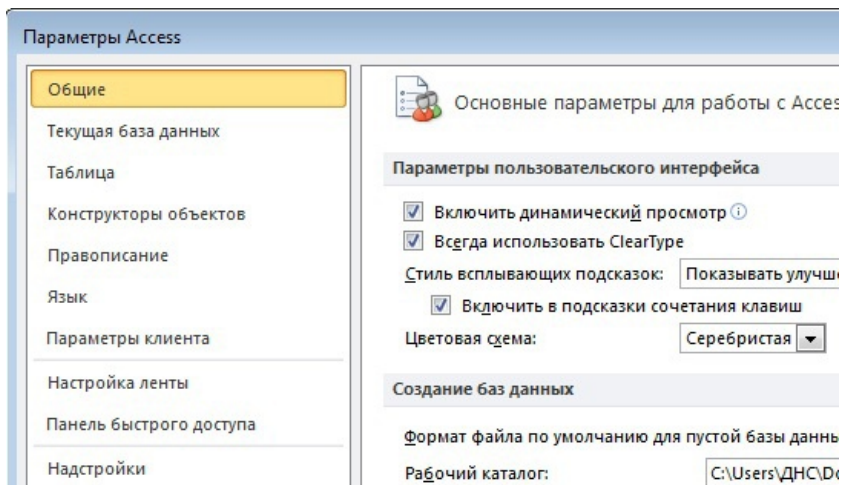


Рис. 92. Параметры Access

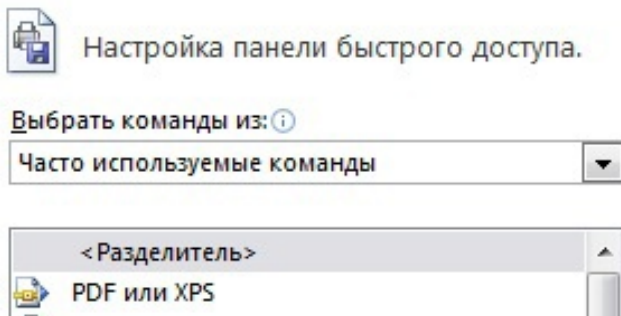


Рис. 93. Выбор набора команд

строку «Вкладка «Работа с базами данных»» (рис. 94). В окне появится набор команд. Выберем «Диспетчер кнопочных форм». Нажмем кнопку «Добавить» справа

от списка (рис. 95). Команда будет добавлена в конец

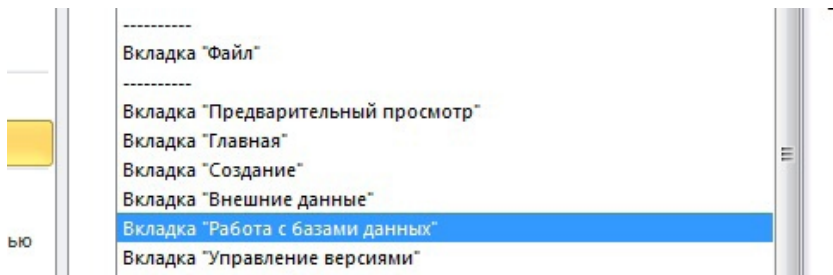


Рис. 94. Выбор команды

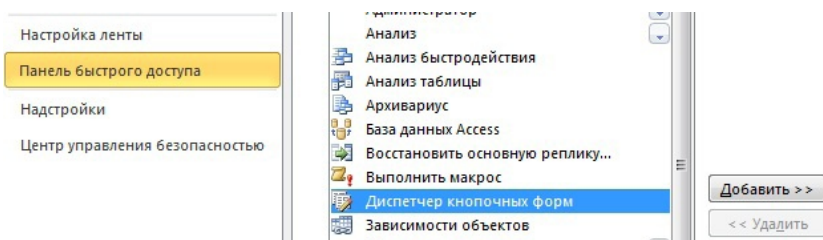


Рис. 95. Добавление новой команды на панель быстрого доступа

списка справа (рис. 96а). Оставим в поле над списком значение по умолчанию «Для всех документов». Нажмем «ОК». В левом верхнем углу окна базы данных появился значок диспетчера кнопочных форм (рис. 96б). Почему этот значок не поставили на панель сразу? Ответ очевиден. Если все команды из только что рассмотренного списка разместить на панели быстрого доступа, панель перестанет быть таковой. А потому каждый сам обу-

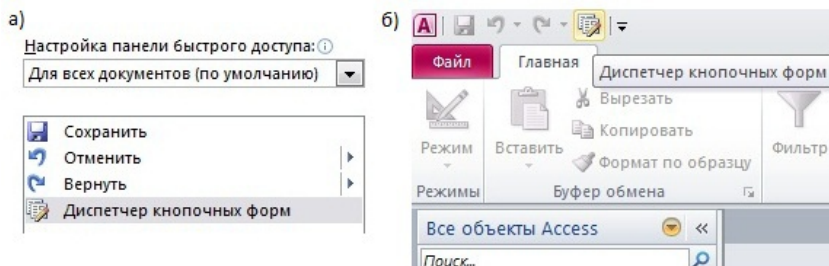


Рис. 96. Установка пиктограммы для кнопочной формы

страивает свое рабочее место так, как ему кажется комфортней. Щелчком по значку диспетчера. Всплывет диалоговое окно, в котором сообщается, что не удалось найти кнопочную форму (рис. 97). Правильно. Ведь мы ее еще

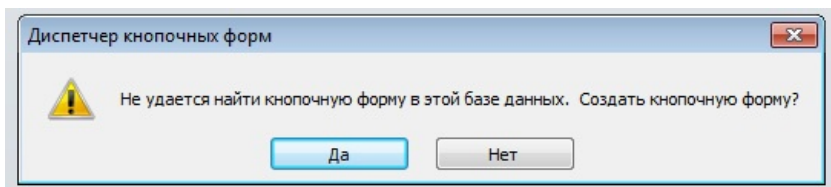


Рис. 97. Создание кнопочной формы

не создали. На вопрос «Создать кнопочную форму?» ответим: «Да». Появится окно с заголовком «Диспетчер кнопочных форм» (рис. 98). Осталось создать и заполнить странички кнопочной формы. Но мы ненадолго отложим эту работу и закроем окно диспетчера нажатием кнопки

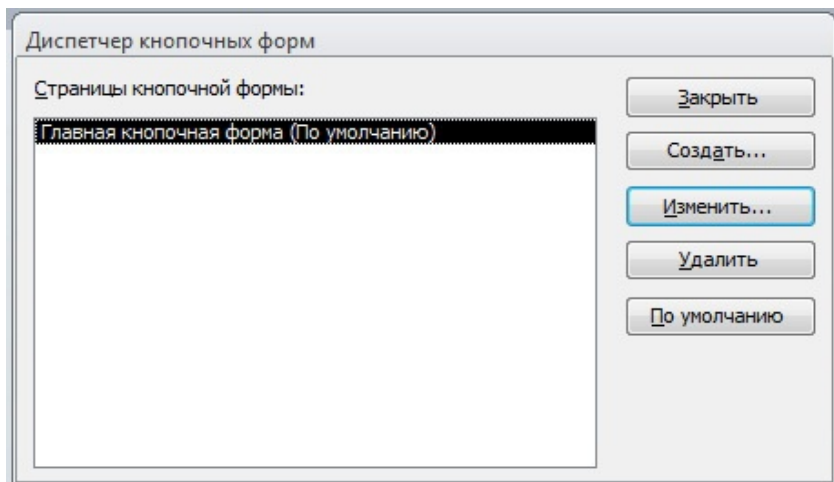


Рис. 98. Окно диспетчера кнопочных форм

«Закреть». В дальнейшем мы всегда легко можем его открыть щелчком по пиктограмме, которую разместили в левой верхней части окна базы данных (см. рис. 96б).

§ 3.2. Один полезный макрос

89 ⇔ 95 Создадим один полезный макрос. В предыдущей главе мы уже имели дело с внедренными макросами, которые привязаны к конкретному объекту и активизируются только при наступлении того события, в соответствие которому их поставили. Сейчас мы создаем так называемый **автономный макрос**. Для этого выберем пункт

меню «Создание» и в группе «Макросы» нажмем кнопку «Макрос» (рис. 99). На экране появится уже знакомый

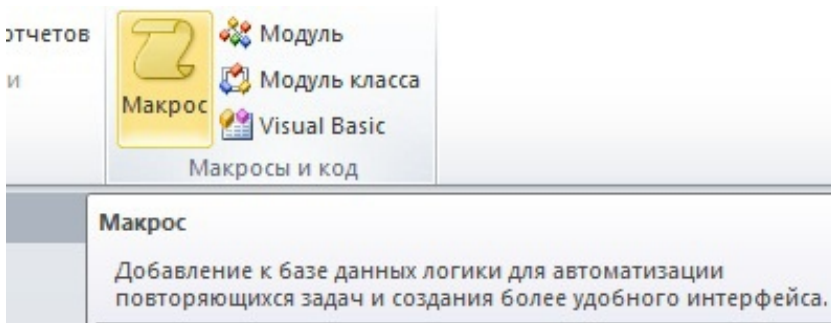


Рис. 99. Создание автономного макроса

нам «Каталог макрокоманд». Выберем в каталоге «Системные команды», а затем «ВыйтиИзAccess» (рис. 100). Наш макрос опять будет состоять из одной единственной

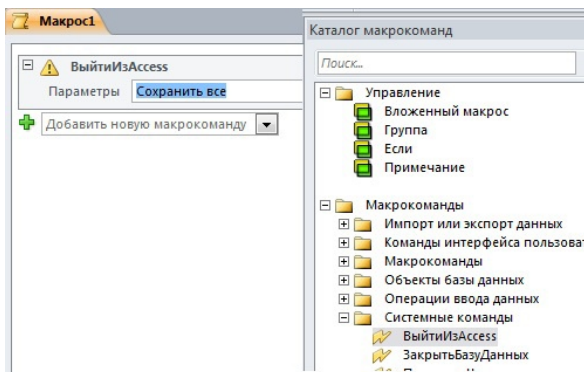


Рис. 100. Макрокоманда «ВыйтиИзAccess»

макрокоманды. Закроем окно и сохраним макрос под именем «Выход» (рис. 101). Читатель, возможно, уже

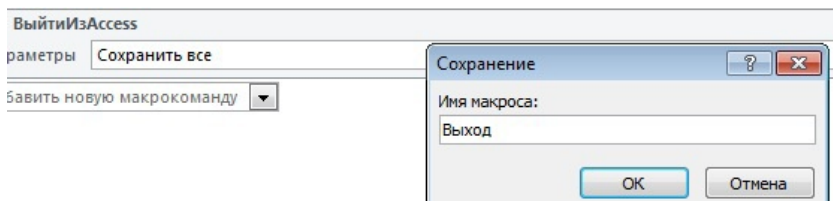


Рис. 101. Сохранение макроса

догадался, что этот макрос будет завершать работу приложения с сохранением результатов.

§ 3.3. Страницы кнопочной формы

93 ⇔ 101 Откроем окно диспетчера кнопочных форм щелчком по пиктограмме (см. рис. 96б). Под надписью «Страницы кнопочной формы» располагается список страниц. Пока мы видим только одну, которой по умолчанию дано имя «Главная кнопочная форма». Нажмем кнопку «Изменить». Под надписью «Название кнопочной формы» запишем новое имя страницы: «Кружки» (рис. 102). «Закрыть». Создадим еще три страницы. Для этого в окне «Страницы кнопочной формы» нажмем «Создать». Под надписью «Название кнопочной формы» запишем имя следующей страницы: «Отчеты». «Закрыть».

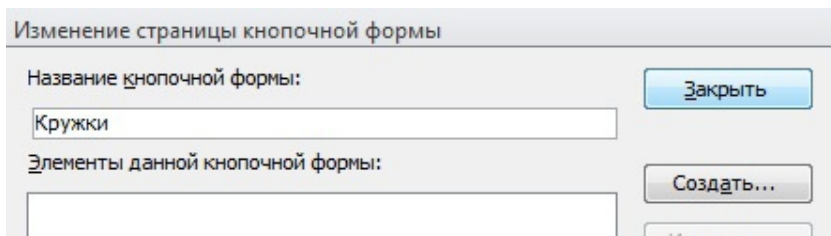


Рис. 102. Новое название страницы

Точно так же создадим страницы «Первичные данные» и «Поиск». Окно примет вид, как на рис. 103. Приступим

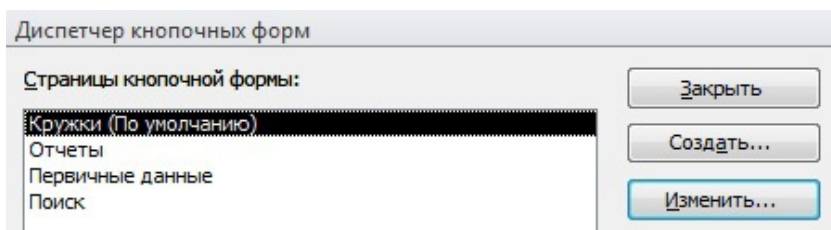


Рис. 103. Список страниц кнопочной формы

к заполнению страниц. В списке выберем главную форму «Кружки» и нажмем «Изменить». Теперь создадим кнопки. Нажмем «Создать». Появится диалоговое окно. В строку «Текст» введем имя кнопки «Первичные данные»; в следующей строке оставим команду «Перейти к кнопочной форме»; в третью строку введем имя кнопочной формы «Первичные данные». Нажмем «ОК» (рис. 104).

В списке «Элементы кнопочной формы» появится

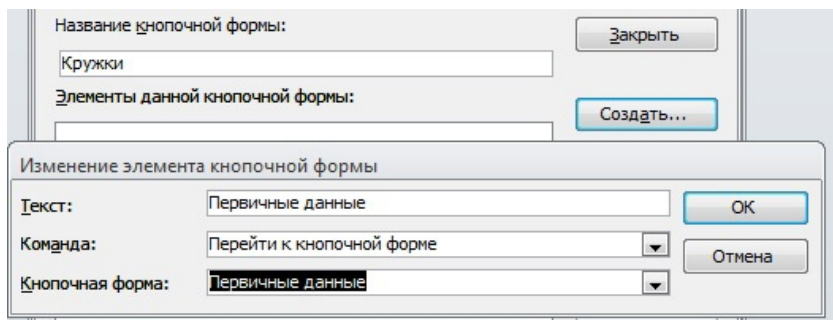


Рис. 104. Заполнение странички «Кружки»

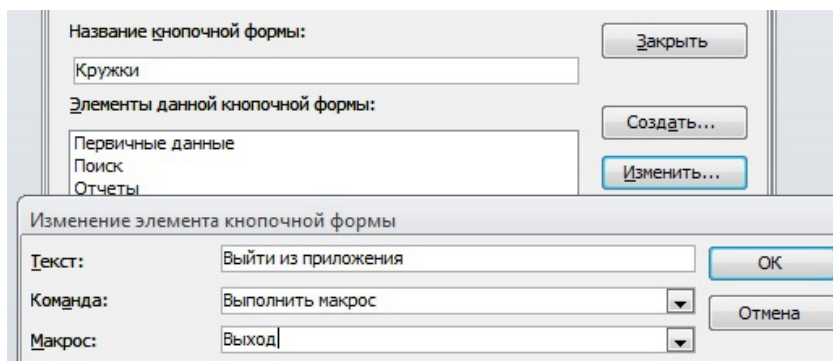


Рис. 105. Создание кнопки «Выйти из приложения»

название кнопки «Первичные данные». Также создадим кнопки «Поиск» для перехода к кнопочной форме «Поиск» и «Отчеты» для перехода к кнопочной форме «Отчеты». Нажмем «Создать». Для этой кнопки введем текст «Выйти из приложения» и выберем команду «Выполнить макрос»; в третьей строке зададим имя созданного на-

ми в предыдущем параграфе автономного макроса «Выход» (см. рис. 105). «ОК». Окно странички «Кружки» приняло вид, как на рис. 106. «Заккрыть». Выберем

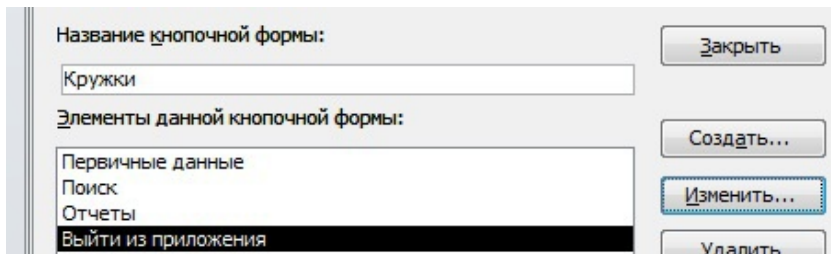


Рис. 106. Страница кнопочной формы «Кружки»

в списке страничек «Первичные данные». «Изменить». Приступим к созданию кнопок. Нажмем «Создать». Введем текст «Список учеников»; выберем команду «Открыть форму для изменения»; зададим имя формы «Список учеников» (рис. 107). «ОК». Аналогично создадим кнопки «Список преподавателей» и «Список кружков». Нажмем «Создать». Для следующей кнопки введем текст «Главное меню». Оставим команду «Перейти к кнопочной форме». Зададим имя странички «Кружки». При нажатии на эту кнопку мы должны вернуться на страничку «Кружки» (рис. 108). «ОК». «Заккрыть». Выберем страничку «Поиск». Создадим кнопки «Поиск ученика», «Поиск преподавателя» и «Поиск кружка» для открытия

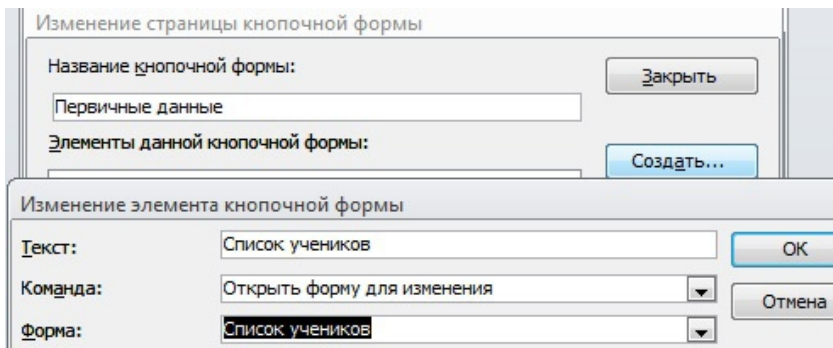


Рис. 107. Кнопка «Список учеников»

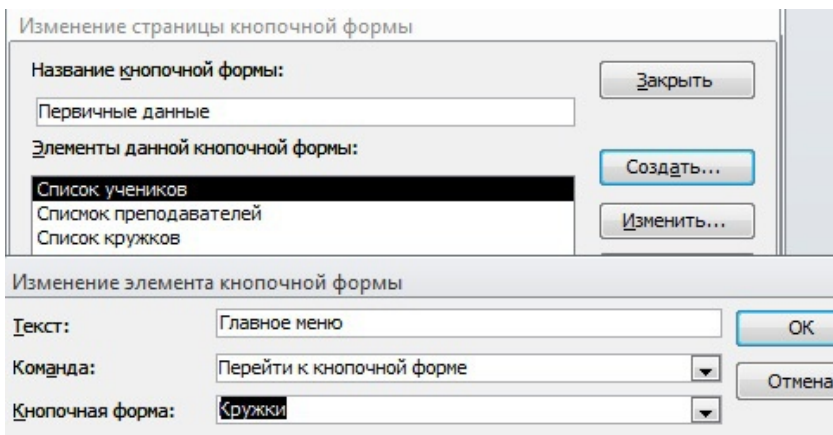


Рис. 108. Страничка «Первичные данные»

одноименных форм. Также добавим кнопку «Главное меню» для перехода на страничку «Кружки» (рис. 109). Нажмем «Закреть» и еще раз «Закреть». Теперь можно начать работу с кнопочной формой. Найдём её

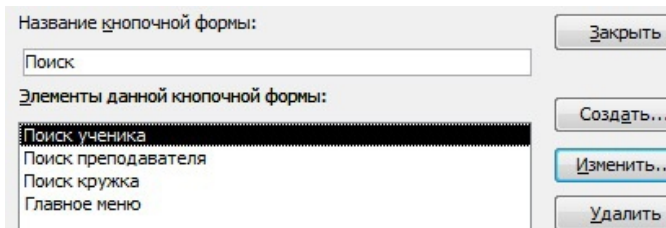


Рис. 109. Список кнопок на страничке «Поиск»

в списке «Все объекты Access» и откроем (рис. 110). Проверим работу кнопочной формы. Убедимся, что при

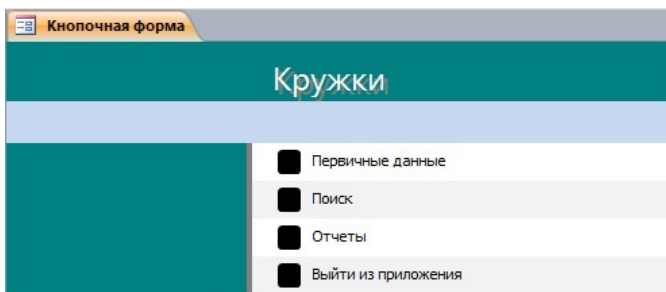


Рис. 110. Кнопочная форма

нажатию кнопок выполняются именно те действия, которые мы хотели предписать этим кнопкам. Кнопочную форму, как и любую другую, можно открыть в режиме конструктора и отредактировать ее внешний вид.

§ 3.4. Параметры запуска

95 ⇔ 102 Теперь работу с базой данных будем начинать с открытия кнопочной формы. А нельзя ли сделать так, чтобы эта форма открывалась сразу автоматически? Для этого надо установить соответствующий параметр базы данных. Откроем вертикальное меню «Файл» и выберем строку «Параметры» (рис. 111а). Откроется окно

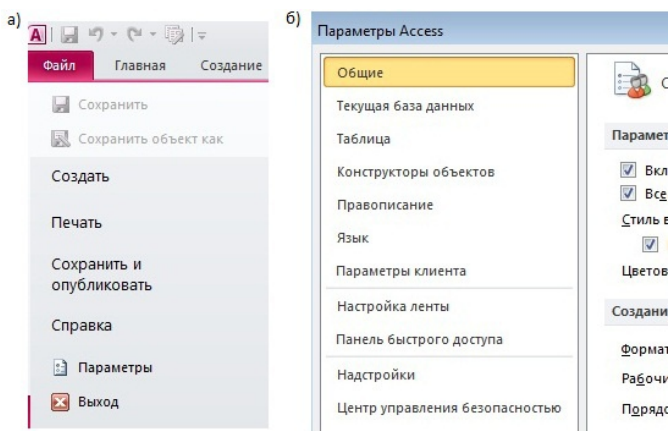


Рис. 111. Параметры Access

«Параметры Access» (рис. 111б). Выберем строку «Текущая база данных». Справа появится панель «Параметры для текущей базы данных» (рис. 112). Развернем меню в строке «Форма просмотра». В списке выберем имя «Кнопочная форма». «ОК». Появится диалоговое окно

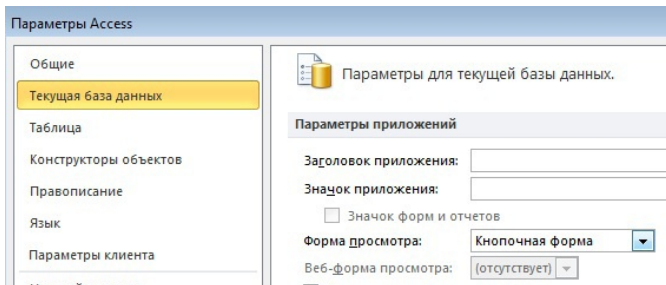


Рис. 112. Параметры для текущей базы данных

с предупреждением о необходимости закрыть и снова открыть базу данных (рис. 113). «ОК».

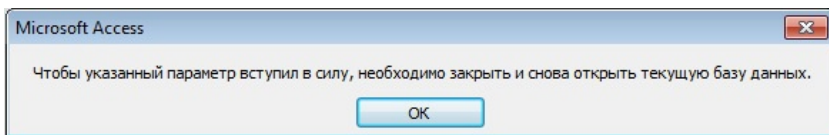


Рис. 113. Диалоговое окно

§ 3.5. Перекрывание окон

101 ⇔ 105 До сих пор, если из формы «Ф1» мы открывали форму «Ф2», она просто располагалась поверх предыдущей формы. Вся последовательность открытых форм отражалась на вкладках сверху окна форм. Однако многие чувствуют себя комфортней, когда последовательно открытые формы располагаются в перекрывающихся

окнах. Чтобы формы располагались таким образом, надо, как и в предыдущем параграфе, открыть панель «Параметры для текущей базы данных» (рис. 114) и в разделе «Параметры окна документа» установить переключатель в положение «Перекрывание окон». Нажмем «ОК». Систе-

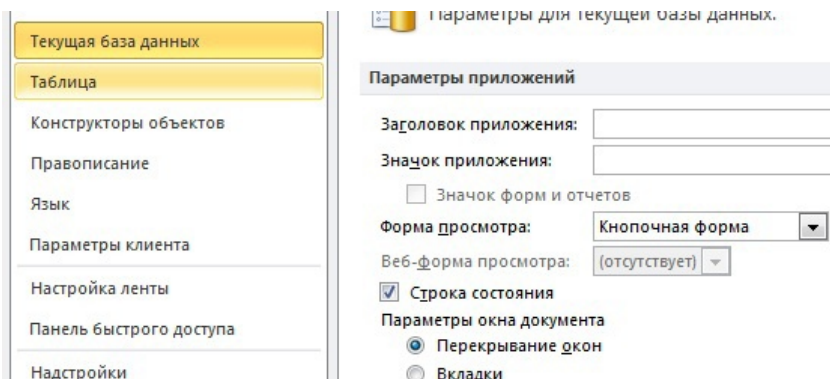


Рис. 114. Параметры окна документа

ма, дабы параметр вступил в силу, опять предложит перезапустить приложение. Теперь последовательно открываемые окна будут перекрывать друг друга (рис. 115). Здесь мы можем столкнуться с одним неудобством. Если новое открытое окно «Ф2» целиком укладывается на экране в область «Ф1» и вы случайно сделали щелчок по «Ф1», то «Ф1» примет фокус и заслонит собой «Ф2». Само по себе это нестрашно, но иногда есть резон запретить работу с ранее открытыми формами до тех пор, пока не закроется

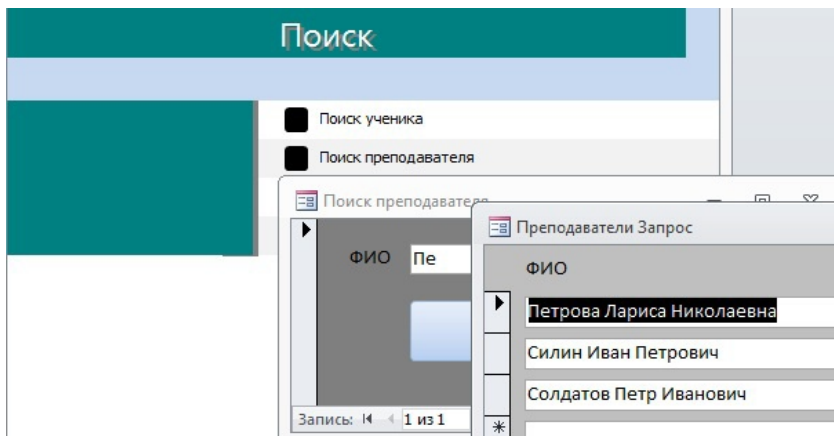


Рис. 115. Перекрывающиеся окна

последняя. Для этого следует поочередно в режиме конструктора сделать все формы модальными (рис. 116). Мы сделали модальной кнопочную форму.

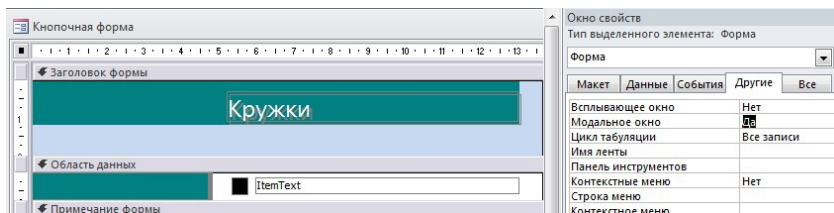


Рис. 116. Установка параметра «Модальное окно»

Теперь нетрудно убедиться, что в процессе работы приложения ранее открытые формы никогда не перекрывают последнюю. В этой главе мы объединили все прежде

разработанные фрагменты базы данных, однако разработка приложения на этом не закончилась.

§ 3.6. Редактирование кнопочной формы

102 ⇔ 113 В кнопочной форме наглядно представлены все возможности созданной нами базы данных и теперь мы можем критически посмотреть на результаты работы. Процесс разработки приложения не состоит из последовательности единственно правильных шагов и не выполняется по какой-то раз и навсегда определенной схеме. В нашем случае мы видим, что пользователь может просмотреть список учеников, занимающихся в данном кружке, но не может рассматривать списки «кружковцев» по классам. Добавим и такую возможность. Для этого создадим новую форму. Специально еще раз разберем весь процесс. Нажмем «Создание», затем «Конструктор форм». Мы не обращаемся к мастеру потому, что форма не связана ни с каким источником данных. На экране появится пустая форма. Возьмем с панели элементов управления элемент «Поле» (рис. 117). В это поле будем записывать номер класса. Для ввода «буквы класса» возьмем «Поле со списком». Откроется диалоговое окно (рис. 118). Выберем фиксированный набор значений. «Далее». Введем

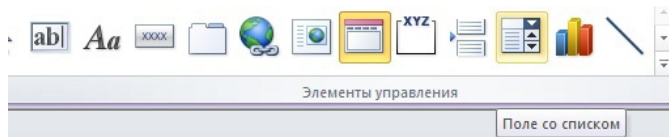


Рис. 117. Панель элементов



Рис. 118. Выбор фиксированного набора значений

список значений: а, б, в, г. Удалим подпись к полю «Буква», передвинем поле со списком ближе к полю с номером класса, чтобы мы могли видеть полностью идентификатор класса. Например, 10а. Первому полю дадим имя «Класс» (рис. 119), а второму – «Буква». Напомним, что каждый

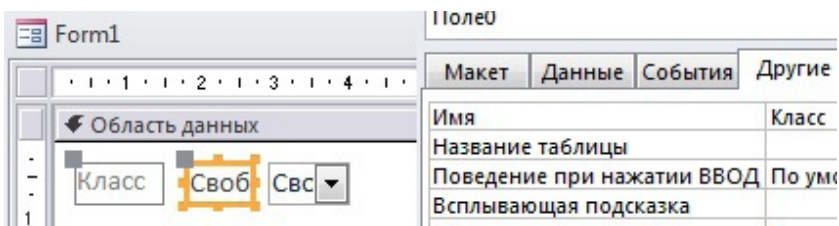


Рис. 119. Присвоение имени элементу управления

элемент управления должен иметь имя, по которому мы могли бы обращаться к нему. Система присвоит элементу имя, даже если мы забудем сделать это. Но с именем типа

«Кнопка17» нам трудно будет работать. Добавим кнопку для закрытия формы. Как и в созданных ранее формах, дадим ей имя «Закрыть» и поставим под ней подпись «Закрыть». Следующая кнопка будет открывать список учеников. Возьмем ее с панели элементов. Здесь мы откажемся от услуг мастера. Для этого можно отключить «Мастер» на панели элементов управления или в первом диалоговом окне нажать «Отмена». На экране появится кнопка. Дадим ей имя «Список» (на вкладке «Другие»), выберем рисунок (на вкладке «Макет»). Под кнопкой поместим подпись «Список». Раскрасим форму так же, как мы это делали ранее, и сохраним под именем «Список класса» (рис. 120). Откроем форму в рабочем режиме. Пока

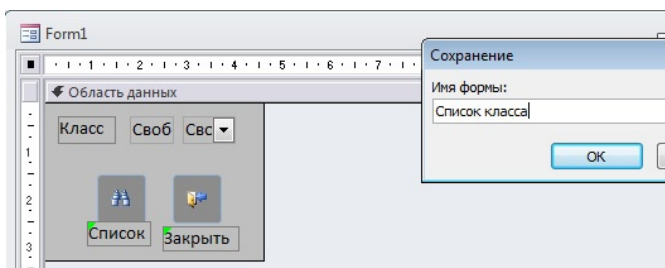


Рис. 120. Сохранение формы

при нажатии на кнопку «Список» ничего не происходит. Работает только кнопка «Закрыть». Заметим, что форма имеет ряд ненужных атрибутов, таких как, например,

область выделения. Переведем форму в режим конструктора. В окне свойств кнопки «Список» перейдем на вкладку «События». Мы уже умеем присваивать событию метод. В данном случае событию «Нажатие кнопки» поставим в соответствие макрос «ОткрытьФорму» (рис. 121). В окне параметров макроса выберем имя формы «Список

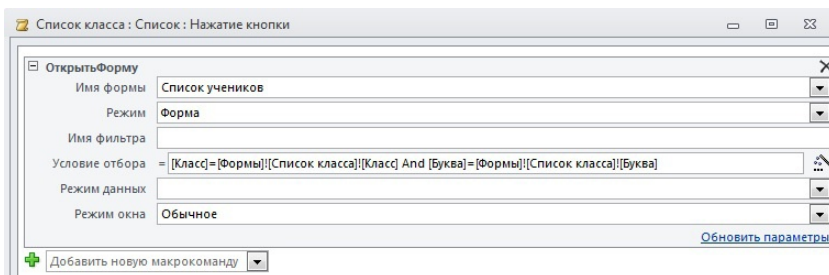


Рис. 121. Параметры макроса

учеников». В строку «Условие отбора» введем текст:

[Класс]=[Формы]![Список классов]![Класс] And
[Буква]=[Формы]![Список классов]![Буква].

Таким образом, мы указали, что критерием отбора записей в форме «Список учеников» будет совпадение значений полей «Класс» и «Буква» со значениями одноименных полей формы «Список класса». Разумеется, совпадение имен полей вовсе не обязательно. Мы могли полям формы «Список класса» дать и другие имена. Сохраним макрос. Перейдем к свойствам формы. На вкладке

«Макет» укажем значения свойств: «Область выделения» – «Нет»; «Кнопки навигации» – «Нет»; «Полосы прокрутки» – «Отсутствуют» (рис. 122). На вкладке «Другие»

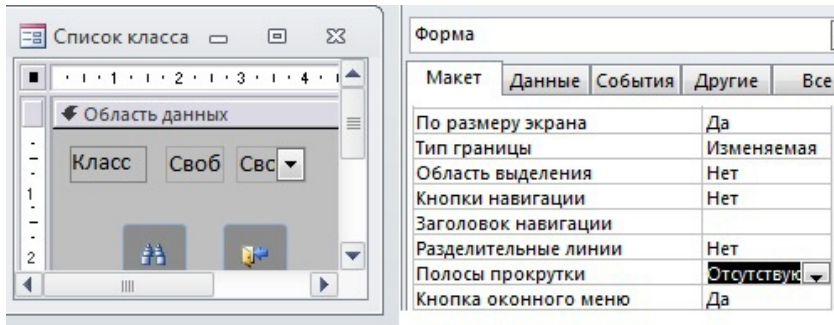


Рис. 122. Свойства формы

сделаем окно модальным. Откорректируем свойства поля «Класс». На вкладке «Данные» в строке «Условие на значение» зададим: ≥ 1 And ≤ 11 (рис. 123). Можно

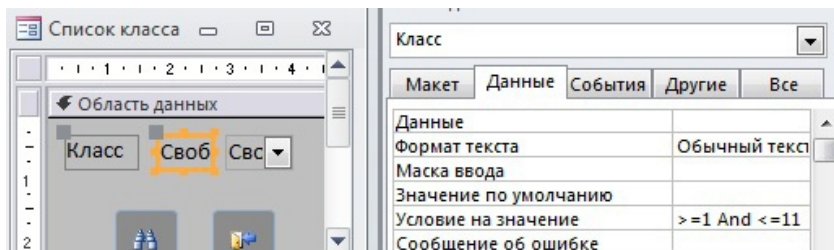


Рис. 123. Установка связи между формами

задать значение по умолчанию, например 10. Для поля

«Класс» также можно задать значение по умолчанию «а». Осталось внести изменения в кнопочную форму. Щелчком по известному нам значку в левой верхней части окна базы данных, откроем страничку «Первичные данные». В списке страниц установим указатель на «Первичные данные» и нажмем «Изменить». На открывшейся страничке нажмем «Создать». Откроется окно для создания кнопки (рис. 124). Введем текст «Список класса».

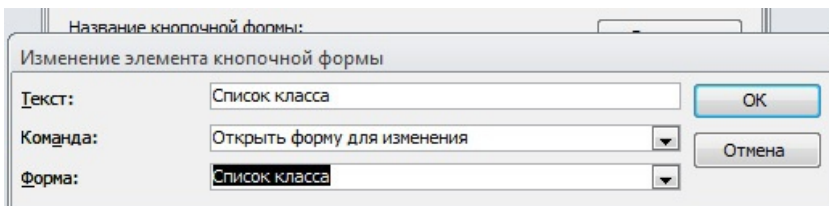


Рис. 124. Создание кнопки

В строке «Команда» выберем «Открыть форму для изменения». В строке «Форма» выберем «Список класса». «OK». Новая кнопка оказалась внизу списка. Установим указатель на ней и, пользуясь кнопкой «Вверх», передвинем в положение после кнопки «Список учеников» (рис. 125). Закроем окно странички, а затем и всего диспетчера кнопочных форм. После всех правок еще раз протестируем приложение. Откроем кнопочную форму. В ней нажмем кнопку «Список класса». Откроется одноименная

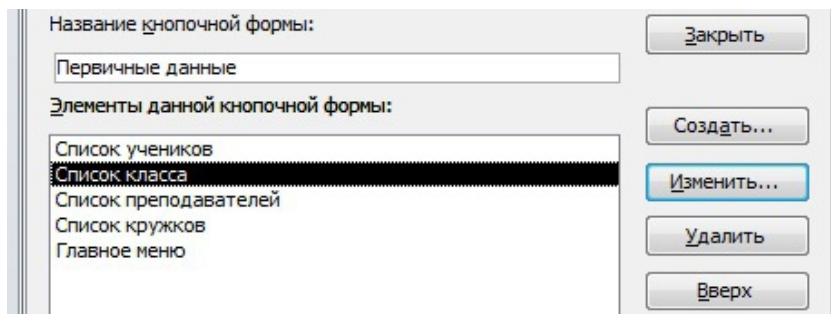


Рис. 125. Задание имени формы

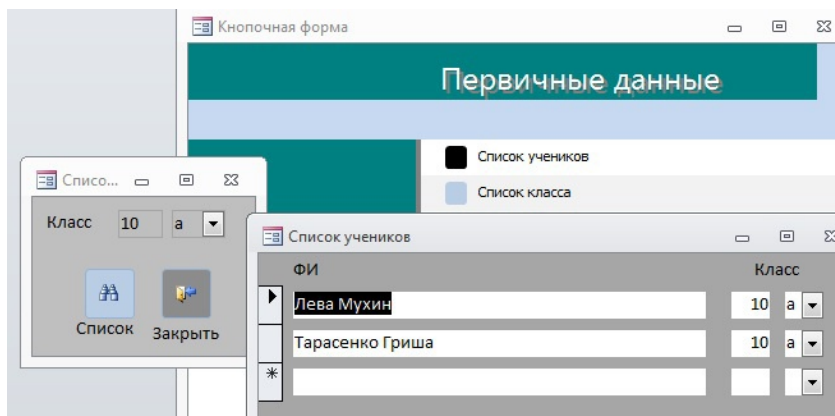


Рис. 126. Список учеников 10а класса

форма. Зададим в поле класс значение 10, а в поле «Буква» – «а» и нажмем кнопку «Список». Появится список всех «кружковцев», учеников 10а класса (рис. 126). В нашем случае их оказалось всего двое, так как мы для тестирования приложения создали небольшой набор

первичных данных. На последнем этапе разработки тестирование желательно провести с большим (а еще лучше с полным) набором данных.

Поскольку в параметрах Access установлено «Перекрытие окон», на экране мы видим сразу все открытые формы. Их положение на экране можно определить, задав свойство на вкладке «Макет» в строке «Выравнивание по центру»: «Да» или «Нет».

Глава 4. Отчеты

В Microsoft Access пользователь имеет возможность вывести на печать содержимое любой таблицы, запроса или формы. Однако согласитесь, в результате получится не то, что можно положить начальнику на стол. Отчет – это документ, предназначенный для печати. В простейшем случае он состоит из заголовка – текста, расположенного в начале документа; верхнего колонтитула – текста, расположенного в верхней части каждой страницы; области данных; нижнего колонтитула – текста, расположенного внизу каждой страницы; примечания – текста, расположенного в конце отчета.

Отчет может иметь и более сложную структуру. Но мы будем двигаться от простого к сложному.

§ 4.1. Отчет «Весь список»

105 ⇔ 118 Прежде всего создадим отчет, который будет представлять собой список всех «кружковцев» в алфавитном порядке с указанием класса, дня рождения и телефона. Для этого выберем пункт меню «Создание» и щелкнем по пиктограмме «Мастер отчетов». В первом диалоговом окне выберем в качестве источника данных таблицу

«Ученики». Возьмем для отчета поля «ФИ», «Класс», «Буква», «ДР» и «Телефон» (рис. 127). «Далее». В следующем

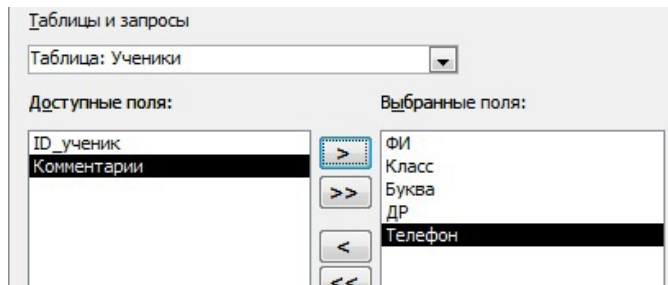


Рис. 127. Выбор таблицы и полей

диалоговом окне мастер предложит добавить уровни группировки. Обойдемся без группировки. «Далее» (рис. 128). Установим сортировку по возрастанию значений поля «ФИ»,

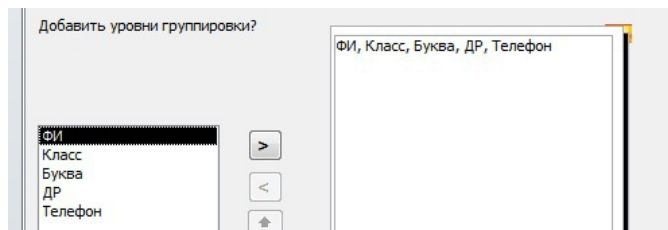


Рис. 128. Уровни группировки

т. е. по алфавиту (рис. 129). «Далее». Выберем макет «табличный» и ориентацию «книжная» (рис. 130). «Далее». Дадим отчету имя «Весь список» (рис. 131). «Готово». На экране появится отчет (рис. 132). Обычно после

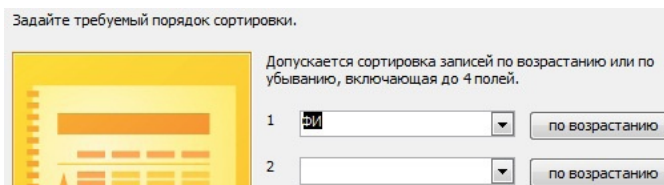


Рис. 129. Сортировка по значениям поля «ФИ»

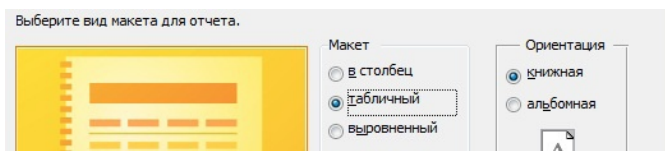


Рис. 130. Выбор макета и ориентации страницы

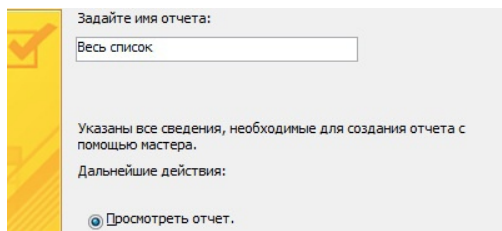


Рис. 131. Сохранение отчета

мастера отчет приходится дорабатывать. В частности, если мы собираемся печатать отчеты на черно-белом принтере, цветной фон в заголовке отчета будет неуместен. Щелчком по значку конструктора в правом нижнем углу окна базы данных переведем отчет в режим конструктора (рис. 133). Точно так же, как в случае с формами, отредактируем цвет фона и текста. После небольших

ФИ	Класс	Буква	День рождения	Телефо
Бурягин Витя	11	а	1 мая 2003 г.	+7(921)
Бучиков Вена	9	а	13 ноября 2005 г.	+7(921)
Демина Света	11	б	11 мая 2004 г.	+7(960)
Иванова Даша	10	б	23 мая 2005 г.	+7(921)
Карлов Саша	11	б	30 апреля 2004 г.	+7(921)
Коровин Сева	9	а	19 октября 2006 г.	+7(921)
Кошкин Вася	9	б	1 января 2006 г.	+7(921)
Крылов Дима	11	а	3 июня 2004 г.	+7(921)
Кузина Катя	10	б	5 мая 2005 г.	+7(911)
Лева Мухин	10	а	2 декабря 2005 г.	+7(921)
Мышкина Марина	10	б	3 февраля 2006 г.	+7(960)
Перепелкина Оля	10	б	28 марта 2005 г.	+7(960)
Петров Коля	9	б	4 июля 2006 г.	+7(911)
Попова Ира	9	а	4 июля 2006 г.	+7(921)
Поголов Федя	11	б	23 марта 2004 г.	+7(921)

Рис. 132. Отчет «Весь список»

Весь список

Заголовок отчета: Весь список

Верхний колонтитул: ФИ | Класс | Буква | День рождения | Телефо

Область данных: ФИ | Класс | Буква | ДР | Телефо

Нижний колонтитул: =Now() | = "Стр. " & [Page] & " из " & [Pages]

Примечание отчета

Рис. 133. Отчет «Весь список» в режиме конструктора

корректировок отчет примет вид, как на рис. 134. Надо заметить, между формами и отчетами много общего. В частности, отчет, как и форма, является контейнером для элементов управления. На рисунке мы видим основные разделы отчета, о которых говорили в начале

Весь список

Верхний колонтитул

ФИ Класс День рождения Телефон

Область данных

ФИ Кла Бу ДР Телефон

Нижний колонтитул

=Now() = "Стр. " & [Page] & " из " & [Pages]

Примечание отчета

Рис. 134. Отчет после корректировки

Весь список

Весь список

ФИ	Класс	День рождения	Телефон
Бурякин Витя	11 а	1 мая 2003 г.	+7(921)004-34-34
Бучиков Вени	9 а	13 ноября 2005 г.	+7(921)001-21-21
Демкина Света	11 б	11 мая 2004 г.	+7(960)008-76-44
Иванова Даша	10 б	23 мая 2005 г.	+7(921)004-49-75
Карпов Саша	11 б	30 апреля 2004 г.	+7(921)003-56-46
Коровин Сева	9 а	19 октября 2006 г.	+7(921)009-18-19
Кошкин Вася	9 б	1 января 2006 г.	+7(921)002-12-12
Крылов Дима	11 а	3 июня 2004 г.	+7(921)008-89-76
Кузина Ната	10 б	5 мая 2005 г.	+7(911)002-31-23
Лева Мухом	10 а	2 декабря 2005 г.	+7(921)003-13-13
Машкина Марина	10 б	3 февраля 2006 г.	+7(960)001-91-91
Перепелкина Оля	10 б	28 марта 2005 г.	+7(960)003-78-78
Петров Коля	9 б	4 июля 2006 г.	+7(911)004-14-14
Полова Ира	9 а	4 июля 2006 г.	+7(921)008-91-23
Попатов Федя	11 б	23 марта 2004 г.	+7(921)009-06-43
Рысаная Роза	10 б	2 апреля 2004 г.	+7(911)007-65-43
Сидоров Семл	9 б	9 июля 2006 г.	+7(911)004-34-34
Сидорова Оля	11 а	4 апреля 2004 г.	+7(911)009-87-65
Синичкина Лена	9 б	1 июня 2006 г.	+7(960)002-12-00
Тарасенко Гриша	10 а	12 марта 2004 г.	+7(960)003-23-23

18 октября 2019 г. Стр. 1 из 1


Рис. 135. Окончательный вид отчета «Весь список»

параграфа. Обратим внимание на поля в области нижнего колонтитула. Поле слева содержит вычисляемое значение «=Now()». Функция Now при обращении к ней возвращает значение текущей даты. Выражение в поле справа формирует текстовую строку с номером текущей страницы отчета и общего количества страниц. Причем номер текущей страницы хранится в поле отчета [Page], а общее количество страниц – в поле [Pages]. Закроем конструктор с сохранением результатов. Теперь при открытии отчет будет иметь вид, как на рис. 135.

§ 4.2. Отчет «Ученики по классам»

113 ⇔ 125 Следующий отчет будет содержать список учеников, сгруппированный по классам. Источник данных – таблица «Ученики». Выберем поля «ФИ», «ДР», «Класс» и «Буква» (рис. 136). Выделим два уровня группировки по значениям полей «Класс» и «Буква» (рис. 137). Внутри групп упорядочим записи по значениям поля «ФИ» (рис. 138). Выберем макет отчета «ступенчатый» и книжную ориентацию страницы (рис. 139). Сохраним отчет под именем «Ученики по классам» (рис. 140). Поскольку в последнем диалоговом окне мы выбрали параметр «Просмотреть отчет», на экране появится только что создан-

Создание отчетов



Выберите поля для отчета.
Допускается выбор нескольких таблиц или запросов.

Таблицы и запросы
Таблица: Ученики

Доступные поля: Выбранные поля:

ID_ученик	>	ФИ
Телефон	>>	ДР
Комментарии		Класс
		Буква

Рис. 136. Выбор источника данных и полей

Добавить уровни группировки?

ФИ	>	Класс
ДР	<	Буква
		ФИ, ДР

Рис. 137. Уровни группировки

Выберите порядок сортировки записей.
Допускается сортировка записей по возрастанию или по убыванию, включающая до 4 полей.

1	ФИ	по возрастанию
2		по возрастанию

Рис. 138. Сортировка записей

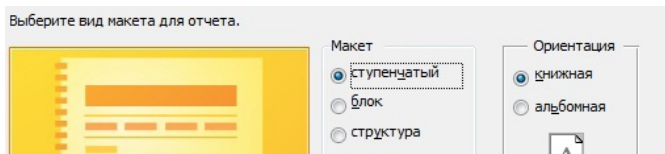


Рис. 139. Выбор макета отчета

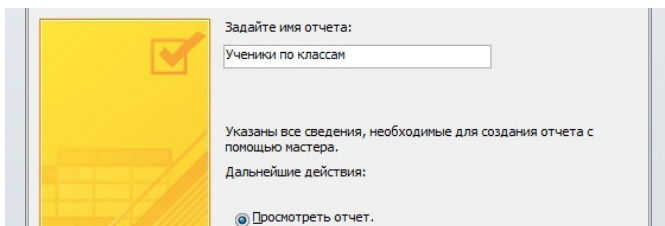


Рис. 140. Сохранение отчета

ный нами документ (рис. 141). Пока работа мастера нас не очень устраивает. Расположение полей в области данных не соответствует надписям. Поля «Класс» и «Буква» захватили большую часть пространства отчета. Да еще у двух учеников вместо дат рождения проставлены последовательности символов «#». Последнее бывает, если данные не укладываются в размер поля в форме или отчете. Переведем отчет в режим конструктора и отредактируем его, как показано на рис. 142. Также добавим линии, разделяющие данные о классах. Эти линии вы легко найдете на панели элементов управления. Закроем конструктор с сохранением. Если теперь открыть отчет

Ученики по классам			
Класс	Буква	ФИ	День рождения
9	a	Бучиков Венья	#####
		Коровин Сева	#####
		Попова Ира	4 июля 2006 г.
б		Кошкин Вася	1 января 2006 г.
		Петров Коля	4 июля 2006 г.
		Сидоров Сеня	9 июля 2006 г.
		Синичкина Лена	1 июня 2006 г.
10	a	Лева Мухин	2 декабря 2005 г.
		Тарасенко Гриша	12 марта 2004 г.

Рис. 141. Отчет, созданный мастером

The screenshot shows a report builder window titled 'Ученики по классам'. The interface includes a header section with the report title, a column header section with fields for 'Класс', 'Буква', 'ФИ', and 'День рождения', and a data grid section with columns for 'Класс', 'Буква', 'ФИ', and 'ДР'. The data grid contains the same data as shown in Figure 141. The interface also includes a footer section with a page number field and a page count field.

Рис. 142. Отчет в режиме конструктора

в рабочем режиме, он будет иметь вид, как на рис. 143. Допустим, нас попросили представить в отчете данные

Ученики по классам			
Класс	Буква	ФИ	День рождения
9			
	а	Бучиков Веня	13 ноября 2005 г.
		Коровин Сева	19 октября 2006 г.
		Попова Ира	4 июля 2006 г.
	б	Кошкин Вася	1 января 2006 г.
		Петров Коля	4 июля 2006 г.
		Сидоров Сеня	9 июля 2006 г.
		Синичкина Лена	1 июня 2006 г.
10			
	а	Лева Мухин	2 декабря 2005 г.
		Тарасенко Гриша	12 марта 2004 г.
	б	Иванова Даша	23 мая 2005 г.

Рис. 143. Отчет «Ученики по классам»

о численности «кружковцев». В режиме конструктора мы видели, что отчет имеет заголовок и примечание, а группам «Класс» и «Буква» мастер выделил только заголовок. Здесь явно есть какая-то дискриминация. На самом деле группам тоже полагается примечание. Наждем в верхней части окна базы данных на пиктограмму «Группировка» (рис. 144а). В нижней части появится панель с надписью «Группировка, сортировка и итоги». В строке

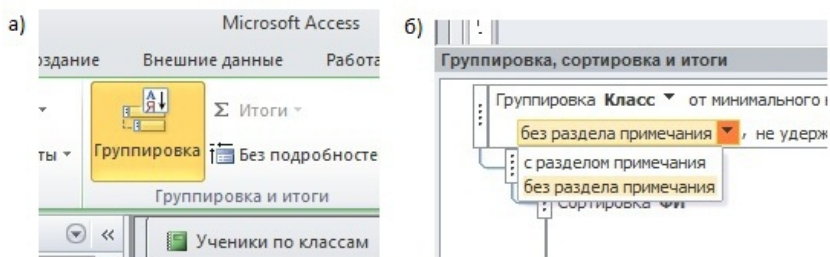


Рис. 144. Создание примечания группы

«Группировка Класс» нажмем «Больше» и развернем меню под текстом «без раздела примечания». Выберем «с разделом примечания» (рис. 144б). То же сделаем и в строке «Группа Буква». Как видно на рис. 145, в окне конструктора появились области примечаний групп «Класс» и «Буква». Теперь настало время объяснить, зачем нам эти области нужны. А нужны они нам затем, что в них мы собираемся разместить данные о численности групп. Поместим в примечание группы «Буква» поле. Поле возьмем с панели элементов. Отформатируем его, как показано на рис. 145. Сделаем надпись «Количество в классе». В поле поместим вычисляемое выражение $=\text{Count}^*$. Введем надпись «Количество в классе». Функция Count^* возвращает количество записей в группе. Скопируем это поле в примечание группы «Класс» и в примечание отчета. Текст в надписи заменим на

☛ Заголовок отчета	
Ученики по классам	
☛ Верхний колонтитул	
Класс	Буква
ФИ	День рождения
☛ Заголовок группы 'Класс'	
Кла	
☛ Заголовок группы 'Буква'	
Бу	
☛ Область данных	
ФИ	ДР
☛ Примечание группы 'Буква'	
Количество в классе	=Count(*)
☛ Примечание группы 'Класс'	
По всем буквам	=Count(*)
☛ Нижний колонтитул	
=Now()	= "Стр. " & [Page] & " из " & [Pages]
☛ Примечание отчета	
Итого	=Count(*)

Рис. 145. Редактирование отчета «Ученики по классам»

Перепелина Оля	28 марта 2005 г.
Рысаков Рома	2 апреля 2004 г.
	Количество в классе
	По всем буквам
11	
а	
Бурягина Витя	1 мая 2003 г.
Крылов Дима	3 июня 2004 г.
Сидорова Оля	4 апреля 2004 г.
	Количество в классе
б	
Демична Света	11 мая 2004 г.
Карпов С аша	30 апреля 2004 г.
Потапов Феда	23 марта 2004 г.
	Количество в классе
	По всем буквам
	Итого
	20
18 октября 2019 г.	Стр. 1 из 1

Рис. 146. Окончательный вид отчета «Ученики по классам»

«По всем буквам» и «Итого» соответственно. Закроем конструктор с сохранением. Если теперь открыть отчет, он будет иметь вид, как на рис. 146.

§ 4.3. Отчет «Возраст учащихся»

118 ⇔ 140 Создадим отчет, который по каждому кружку будет выдавать средний возраст, минимальный возраст и максимальный возраст посещающих его учеников, а также их численность. Для этого нам понадобится источник данных – запрос. Выберем пункт меню «Создание» и щелкнем по пиктограмме «Мастер запросов». В первом диалоговом окне выберем строку «Простой запрос» (рис. 147). «Далее». Выбираем таблицу «Ученики» и ее поля

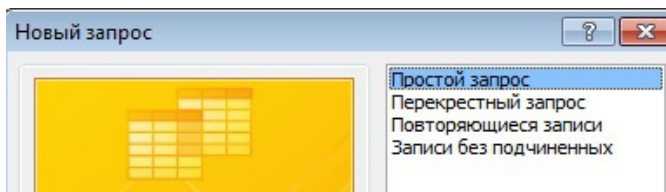


Рис. 147. Простой запрос

«ID_ученик», «ФИ», «ДР» (рис. 148). «Далее». Сохраним запрос под именем «Возраст». В рабочем режиме он будет иметь вид, как на рис. 149. Переведем запрос в режим конструктора (рис. 150). Мы видим три заполненных столбца. В первый свободный столбец справа в строке «Поле» введем текст: `Возраст: DateDiff("уууу";[ДР];Date())`. Таким образом, мы создали в запросе поле с вычисляемым значением. До двоеточия – имя поля – «Возраст».

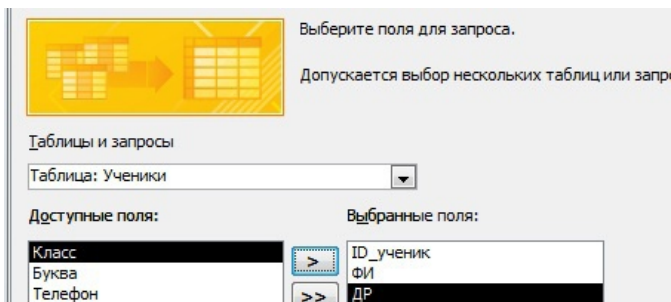


Рис. 148. Выбор источника данных и полей

ID_ученик	ФИ	День рождения
1	Кошкин Вася	1 января 2006 г.
2	Тарасенко Гриша	12 марта 2004 г.
3	Лева Мухин	2 декабря 2005 г.
4	Кузина Катя	5 мая 2005 г.
5	Бучиков Веня	13 ноября 2005 г.

Рис. 149. Запрос в рабочем режиме

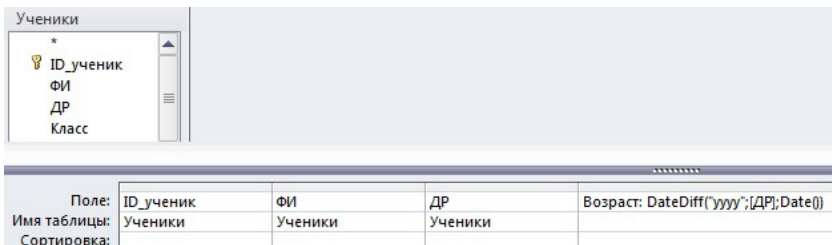
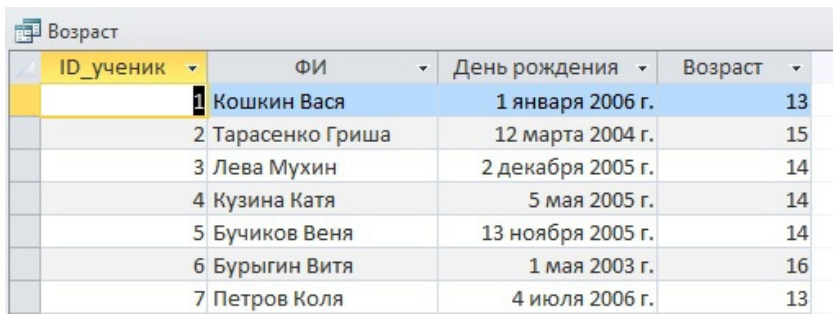


Рис. 150. Запрос «Возраст» в режиме конструктора

То, что после двоеточия, вычисляемое значение, которое находит функция DateDiff. Эта функция возвращает разность начальной и системной, т. е. текущей, даты. Первый

параметр задает единицу времени. В нашем случае "ууу" означает, что разность дат будет выдана в годах. Второй – начальная дата, день рождения ученика [ДР]. Третий – системная дата, которую выдает функция без параметров Date(). Параметры разделены знаком «;». Закроем конструктор с сохранением и откроем запрос в табличном режиме (рис. 151). Как мы видим, действитель-



ID_ученик	ФИ	День рождения	Возраст
1	Кошкин Вася	1 января 2006 г.	13
2	Тарасенко Гриша	12 марта 2004 г.	15
3	Лева Мухин	2 декабря 2005 г.	14
4	Кузина Катя	5 мая 2005 г.	14
5	Бучиков Веня	13 ноября 2005 г.	14
6	Бурьгин Вита	1 мая 2003 г.	16
7	Петров Коля	4 июля 2006 г.	13

Рис. 151. Запрос «Возраст» в рабочем режиме

но появился новый столбец «Возраст». Значение возраста будет меняться каждый раз, когда у кого-то из учеников наступит день рождения. Здесь мы снова видим проявление принципа нормирования данных: не следует хранить в базе то, что может быть получено из уже имеющихся данных. В данном случае неразумно было бы создавать и заполнять поле «Возраст». Следующий запрос. Источником данных для него будут две таблицы и один запрос.

«Создание». «Мастер запросов». В первом диалоговом окне мастера опять выберем строку «Простой запрос». «Далее». Источник данных – таблица «Кружки». Выбранные поля: «ID_кружок» и «Название» (рис. 152).

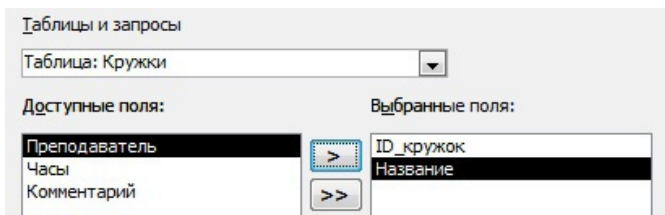


Рис. 152. Поля, выбранные из таблицы «Кружки»

Снова откроем меню «Таблицы и запросы» и выберем таблицу «Ученик_кружок». Перебросим в раздел «Выбранные поля» все ее поля: «ID_ученик_кружок», «Ученик» и «Кружок» (рис. 153). Еще раз откроем меню «Таб-

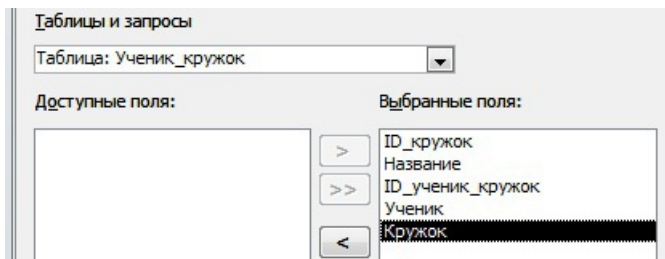


Рис. 153. Поля, добавленные из таблицы «Ученик_кружок»

лицы и запросы», выберем запрос «Возраст». Тот самый,

который только что создали. Из этого запроса возьмем одно поле – «Возраст» (рис. 154). Список полей запроса

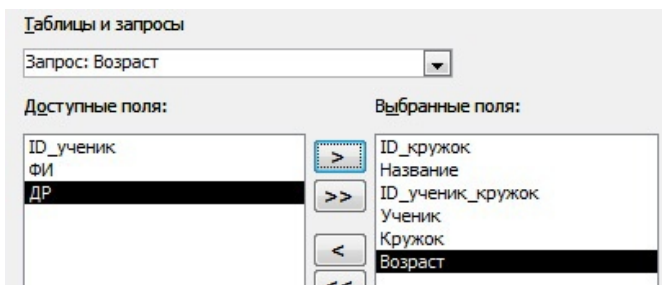


Рис. 154. Поля, добавленные из запроса «Возраст»

сформирован. Нажмем «Далее». Оставим переключатель в положении «подробный» (рис. 155).

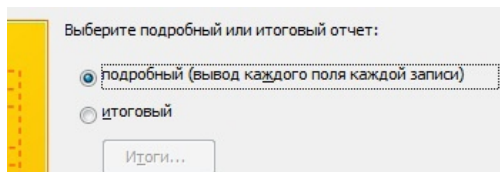


Рис. 155. Запрос «подробный»

Сохраним запрос под именем «Кружки_ученики» (рис. 156). На экране появится подробный запрос, в котором каждой

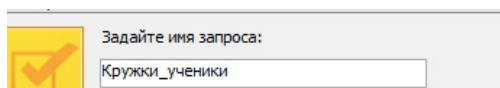
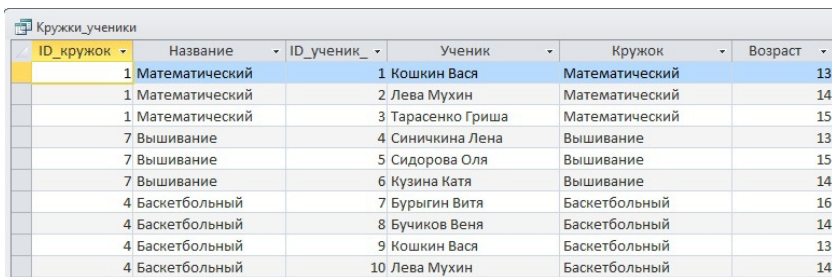


Рис. 156. Сохранение запроса

строке соответствует пара «кружок – ученик» (рис. 157). Закроем запрос. Создадим еще один простой запрос.



ID_кружок	Название	ID_ученик	Ученик	Кружок	Возраст
1	Математический	1	Кошкин Вася	Математический	13
1	Математический	2	Лева Мухин	Математический	14
1	Математический	3	Тарасенко Гриша	Математический	15
7	Вышивание	4	Синичкина Лена	Вышивание	13
7	Вышивание	5	Сидорова Оля	Вышивание	15
7	Вышивание	6	Кузина Катя	Вышивание	14
4	Баскетбольный	7	Бурыгин Витя	Баскетбольный	16
4	Баскетбольный	8	Бучиков Веня	Баскетбольный	14
4	Баскетбольный	9	Кошкин Вася	Баскетбольный	13
4	Баскетбольный	10	Лева Мухин	Баскетбольный	14

Рис. 157. Запрос «Кружки_ученики»

Источник данных – «Кружки_ученики». Выбранные поля: «ID_кружок», «Название» и «Возраст» (рис. 158). На этот раз установим переключатель в положение

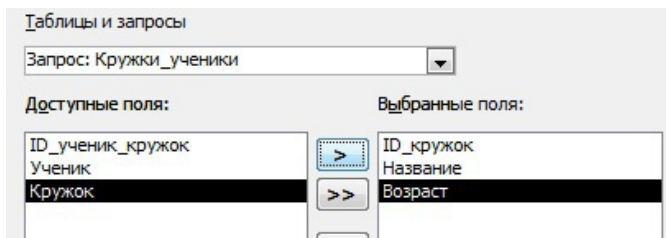


Рис. 158. Выбранные поля

«итоговый» (рис. 159). Нажмем кнопку «Итоги». Появилось окно, в котором мы должны указать, какие итоговые значения следует вычислить (рис. 160). В этом окне мы видим только строку «Возраст». По значениям поля

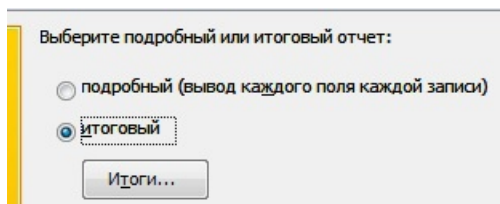


Рис. 159. Итоговый запрос

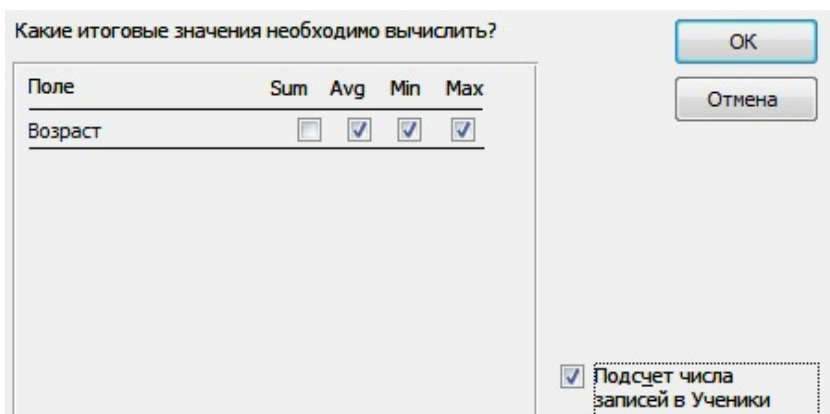


Рис. 160. Выбор итоговых значений

«ID_кружок» записи будут разбиваться на группы. В строке «Возраст» расставим галочки около интересующих нас функций. Выберем Avg – среднее значение, Min – наименьшее значение, Max – наибольшее значение. Кроме того, попросим мастера подсчитать число записей в каждой группе. «ОК». Сохраним запрос под именем «Возраст_учащихся» (рис. 161). На экране появится

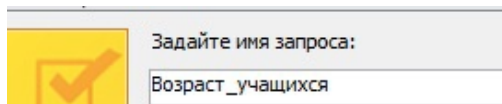


Рис. 161. Сохранение запроса

табличное представление запроса (рис. 162). Рассмотрим

ID_кружок	Название	Avg - Возраст	Min - Возра	Max - Возра	Count - Учени
1	Математический	14	13	15	4
2	Физический	14	13	15	4
3	Моделирования	13,6666666666667	13	15	3
4	Баскетбольный	14,25	13	16	4
5	Химический	14,5	13	15	4
6	Умелые руки	14,25	13	15	4
7	Вышивание	14	13	15	3

Рис. 162. Табличное представление запроса «Возраст_учащихся»

его внимательно и выявим все, что нам здесь не нравится. Нам не нравятся имена, которые присвоила полям запроса система. И еще система выдала средний возраст члена кружка «Моделирование» с точностью до 13-го знака после запятой. Это уж точно лишнее. Переведем запрос в режим конструктора (рис. 163). Рассмотрим верхнюю

ID_кружок	Название	Avg - Возраст: Возраст	Min - Возраст: Возраст	Max - Возраст: Возраст	Count - Ученики: Count(*)
Кружки_ученики	Кружки_ученики	Кружки_ученики	Кружки_ученики	Кружки_ученики	Кружки_ученики
Группировка	Группировка	Avg	Min	Max	Выражение

Рис. 163. Запрос «Возраст_учащихся» в режиме конструктора

строку. Она содержит имена полей. Первые два имени достались от таблицы «Ученики». Следующие поля

вычисляемые. Здесь до знака «:» стоят присвоенные системой имена. После двоеточия в третьем, четвертом и пятом столбцах – имя поля – источника данных для групповых операций. Это поле «Возраст». В последнем столбце после двоеточия видим уже знакомую нам функцию Count(*). Рассмотрим третью строку. В первых двух столбцах стоит «Группировка». Это значит, что значения соответствующих полей положены в основу группировки. В следующих трех столбцах указаны групповые операции. И наконец, в последнем столбце слово «Выражение» говорит нам о том, что значение этого поля вычисляется. Исправим названия полей, начиная с третьего, на «Средний возраст», «Минимальный возраст», «Максимальный возраст» и «Количество» (рис. 164). Займемся форма-

Средний возраст: Возраст	Минимальный возраст: Возраст	Максимальный возраст: Возраст	Количество Count(*)
Кружки_ученики	Кружки_ученики	Кружки_ученики	
Avg	Min	Max	Выражение

Рис. 164. Редактирование запроса «Возраст_ учащихся»

том поля, содержащего средний возраст члена кружка. Для этого в соответствующем столбце щелчком правой клавиши мыши развернем вертикальное меню и выберем в нем строку «Свойства» (рис. 165). В окне свойств установим фиксированный формат поля и число знаков после запятой – 1 (рис. 166). Последнему полю дадим имя

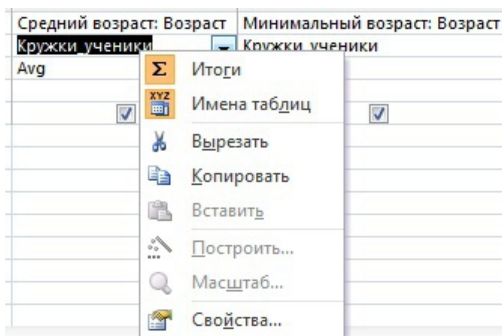


Рис. 165. «Свойства» поля «Средний возраст»

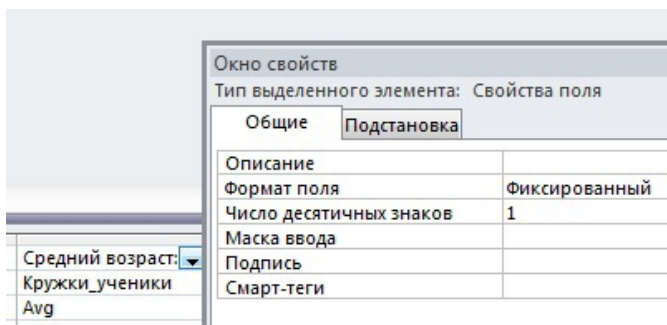


Рис. 166. Формат поля «Средний возраст»

«Количество». Закроем конструктор с сохранением результатов. На экране появится представление отчета в табличном режиме (рис. 167). Закроем запрос. Выберем пункт меню «Создание», щелкнем по пиктограмме «Мастер отчетов». В первом диалоговом окне выберем источник данных – запрос «Возраст_учащихся» и поля «Название», «Средний возраст», «Минимальный возраст», «Максимальный

ID_кружок	Название	Средний возраст	Минимальный возраст	Максимальный возраст	Количество
1	Математический	14,0	13	15	4
2	Физический	14,0	13	15	4
3	Моделирования	13,7	13	15	3
4	Баскетбольный	14,3	13	16	4
5	Химический	14,5	13	15	4
6	Умелые руки	14,3	13	15	4
7	Вышивание	14,0	13	15	3

Рис. 167. Запрос «Возраст учащихся»

возраст» и «Количество» (рис. 168). «Далее». Добавлять

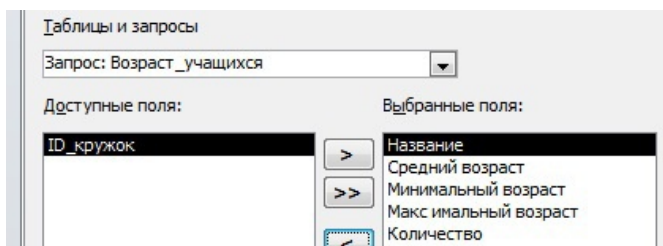


Рис. 168. Выбор источника данных и полей

уровни группировки не будем (рис. 169). «Далее».

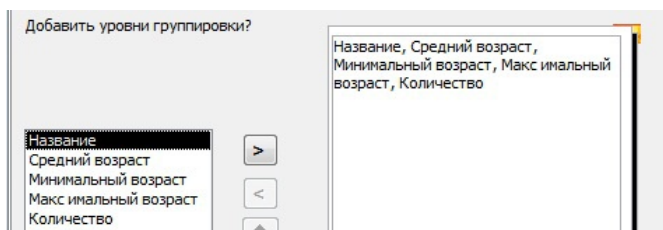


Рис. 169. Уровни группировки

Установим сортировку по возрастанию значений поля «Название» (рис. 170). «Далее». Выберем макет отчета

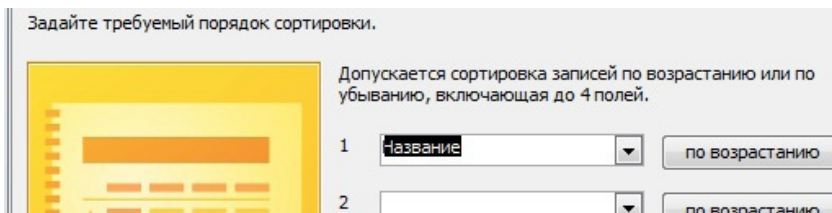


Рис. 170. Сортировка

и ориентацию страницы (рис. 171). Сохраним отчет под



Рис. 171. Макет отчета

именем «Возраст учащихся» (рис. 172). На экране

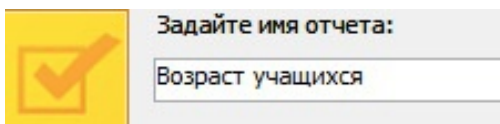


Рис. 172. Сохранение отчета

появится отчет (рис. 173). Переведем его в режим конструктора и отредактируем, как мы делали это ранее, с расчетом на черно-белый принтер. Кроме того, в примечании отчета подведем черту (черту берем с панели элементов управления) и скопируем туда же строку

Название	Средний возраст	Минимальный возраст	Максимальный возраст	Количество
Баскетбольный	14,3	13	16	4
Вышивание	14,0	13	15	3
Математический	14,0	13	15	4
Моделирования	13,7	13	15	3
Умелые руки	14,3	13	15	4
Физический	14,0	13	15	4
Химический	14,5	13	15	4

Рис. 173. Отчет «Возраст учащихся»

области данных. Копируем всю строку потому, что так быстрее, нежели поочередно «таскать» поля с панели элементов. Введем текст надписи «По всем кружкам». Четыре поля будут содержать вычисляемые значения. В первом будет находиться значение среднего возраста по всем кружкам (рис. 174). Среднее от средних. Выберем в окне

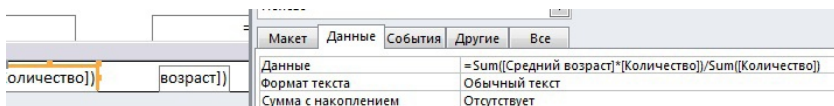


Рис. 174. Окончательный вид отчета «Весь список»

свойств вкладку «Данные» и в строку «Данные» введем текст:

$$=Sum([Средний\ возраст]*[Количество])/Sum([Количество]).$$

Функция Sum возвращает сумму значений поля в той

группе области данных, к которой относится примечание. В нашем случае, поскольку мы работаем в примечании отчета, суммирование производится по всей области данных. Обратите внимание, что значение среднего возраста вычисляется по формуле средней взвешенной. Здесь нельзя просто взять среднее арифметическое, так как численности групп или, как говорят в статистике, веса групп различные. А потому находим сумму произведений значений среднего возраста на количество учеников в кружке: $\text{Sum}([\text{Средний возраст}] * [\text{Количество}])$ и делим ее на суммарное количество учеников во всех кружках $\text{Sum}([\text{Количество}])$. Аналогично устанавливаем вычисляемое значение для минимального минимума: $=\text{Min}([\text{Минимальный возраст}])$ (рис. 175) и для мак-



Рис. 175. Минимальный минимум



Рис. 176. Максимальный максимум

симального максимума: $=\text{Max}([\text{Максимальный возраст}])$ (рис. 176). Общее количество учеников находится просто как сумма количеств по отдельным кружкам (рис. 177):

=Sum([Количество]). Закроем конструктор с сохранением



Рис. 177. Общее количество «кружковцев»

результатов. Теперь отчет будет иметь вид, как на рис. 178. Для создания этого отчета нам пришлось предварительно

Возраст учащихся

Название кружка	Средний возраст	Минимальны	Макс имальны	Количество
Баскетбольный	14,3	13	16	4
Вышивание	14,0	13	15	3
Математический	14,0	13	15	4
Моделирования	13,7	13	15	3
Умелые руки	14,3	13	15	4
Физический	14,0	13	15	4
Химический	14,5	13	15	4
По всем кружкам	14,1	13	16	26

21 октября 2019 г.

Стр. 1 из 1

Рис. 178. Окончательный вид отчета «Возраст учащихся»

создать три запроса. Для каждого, начиная со второго, источником данных является предыдущий запрос.

§ 4.4. Отчет «Ученики по кружкам»

125 ⇔ 148 В § 4.2 мы оформили отчет как список учеников, сгруппированных по классам. Теперь создадим список учеников, сгруппированных по кружкам. Можно, как в § 4.3, сформировать последовательность запросов, а затем мастер создаст отчет с группировкой по полю «Кружок». Но мы, дабы не повторяться, пойдем другим путем. Вспомним, как в § 2.7 мы использовали подчиненную форму (с. 80). Сейчас аналогичные манипуляции проделаем с отчетом. Прежде всего создадим подчиненный отчет. Выберем пункт меню «Создание» и щелкнем по пиктограмме «Мастер отчетов». В первом диалоговом окне выберем таблицу «Ученик_кружок», а в ней поля «Ученик» и «Кружок» (рис. 179). «Далее». Уровни группировки

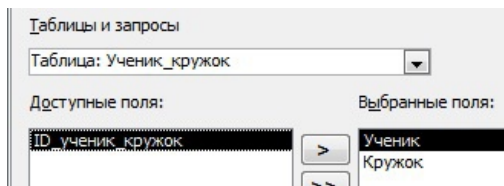
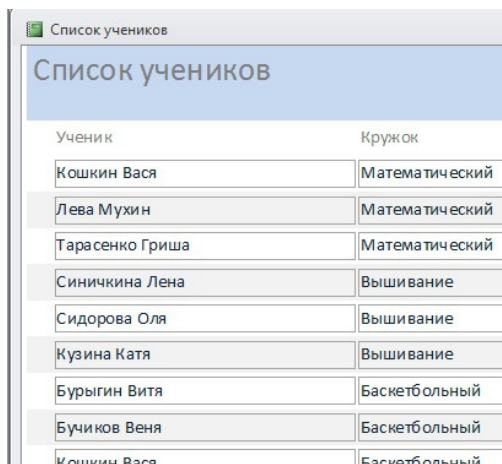


Рис. 179. Выбор таблицы и полей

добавлять не надо. «Далее». Сортировка не требуется. «Далее». Оставляем, обычно принятые по умолчанию, установки: макет «табличный» и ориентация страницы

«книжная». Сохраним отчет под именем «Список учеников». На экране появится отчет в рабочем виде (рис. 180). Переведем его в режим конструктора и внесем ряд исправ-



Ученик	Кружок
Кошкин Вася	Математический
Лева Мухин	Математический
Тарасенко Гриша	Математический
Синичкина Лена	Вышивание
Сидорова Оля	Вышивание
Кузина Катя	Вышивание
Бурьгин Витя	Баскетбольный
Бучиков Веня	Баскетбольный
Кошкин Вася	Баскетбольный

Рис. 180. Отчет «Список учеников»

лений: отредактируем цвета фона и текста; уберем надпись к полю «Кружок», а само поле сделаем невидимым (для этого в свойствах поля на вкладке «Макет» в строке «Вывод на экран» надо установить значение «Нет»); удалим надпись в заголовке отчета; сдвинем до предела вверх планку с надписью «Верхний колонтитул»; в нижнем колонтитуле удалим поля с датой и номером страницы; поднимем до предела вверх планку с надписью «Примечание отчета», как показано на рис. 181. Закроем конструктор

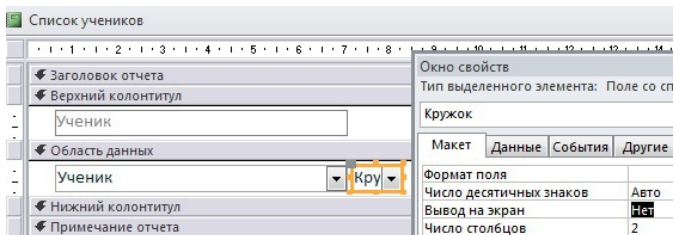


Рис. 181. Свойства поля «Кружок»

с сохранением корректировок. Откроем отчет в рабочем режиме. Теперь он имеет вид, как на рис. 182. Обратим

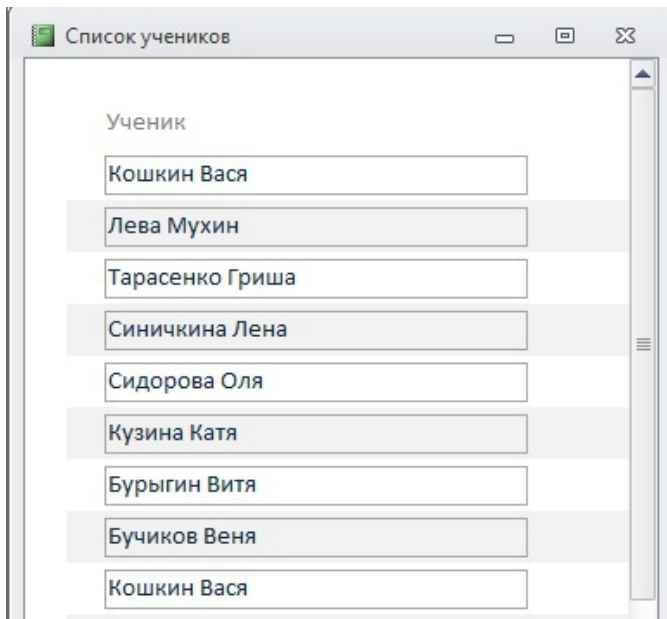


Рис. 182. Окончательный вид отчета «Список учеников»

внимание, что сейчас в отчете содержимое полей может повторяться. Это потому, что на самом деле в списке представлен перечень всех возможных пар «Ученик_кружок». Но поле «Кружок» мы сделали невидимым. В дальнейшем в подчиненной форме оно нам не потребуется, но просто убрать его нельзя, поскольку связь между основной и подчиненной формами будет установлена именно по полю «Кружок».

Теперь создадим основной отчет. В первом окне диалога с мастером выберем источник данных – запрос «Кружки Запрос», а в нем поля «ID_кружок», «Название» и «Преподаватель» (рис. 183). Будем нажимать кнопку «Далее»

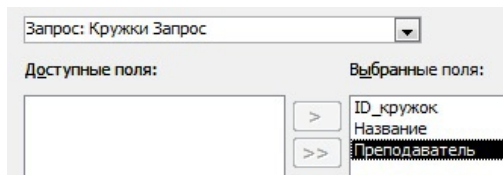


Рис. 183. Выбор запроса и полей

до тех пор, пока мастер не предложит сохранить отчет. Сохраним его под именем «Ученики по кружкам» (рис. 184). На экране появится отчет (рис. 185). Переведем его в режим конструктора и перейдем к редактированию. Отодвинем вниз планку с надписью «Нижний колонтитул» и поместим туда взятый с панели

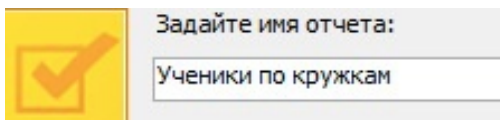


Рис. 184. Сохранение отчета

Ученики по кружкам		
Название	ID_кружок	Преподаватель
Баскетбольный	4	Солдатов Петр Иванович
Вышивание	7	Попова Светлана Борисовна
Математический	1	Петрова Лариса Николаевна
Моделирования	3	Силин Иван Петрович
Умелые руки	6	Крылов Николай Кузьмич
Физический	2	Силин Иван Петрович
Химический	5	Фролова Тамара Николаевна

Рис. 185. Отчет «Ученики по кружкам»

элементов подчиненный отчет (рис. 186). В открывшемся диалоговом окне оставим переключатель в положении «Имеющиеся отчеты и формы» и выберем созданный в этом параграфе подчиненный отчет «Список учеников» (рис. 187). Выберем поле связи «ID_кружок» (рис. 188). Дадим подчиненному отчету имя «Список учеников» (рис. 189). Откорректируем цвета фона и текста, а также размеры полей. Закроем конструктор с сохранением результатов. Теперь отчет выглядит, как на рис. 190.

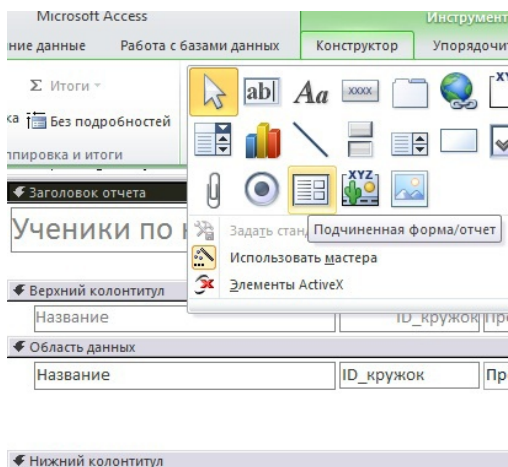


Рис. 186. Отчет «Ученики по кружкам» в режиме конструктора

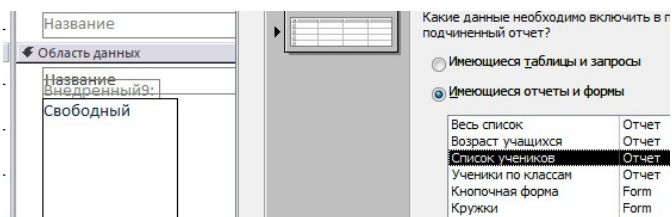


Рис. 187. Выбор подчиненного отчета

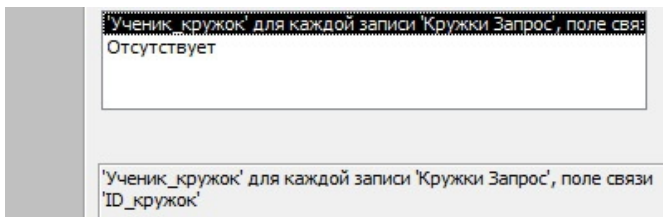


Рис. 188. Выбор поля связи

Задайте имя для подчиненной формы или отчета:

Указаны все сведения, необходимые для создания подчиненной формы или отчета.

Рис. 189. Присвоение имени подчиненной форме

Ученики по кружкам

Ученики по кружкам

Название	Преподаватель
Баскетбольный	<input type="text" value="Солдатов Петр Иванович"/>
Список учеников	
<input type="text" value="Бурыгин Витя"/>	
<input type="text" value="Бучиков Веня"/>	
<input type="text" value="Кошкин Вася"/>	
<input type="text" value="Лева Мухин"/>	
Вышивание	<input type="text" value="Попова Светлана Борисовна"/>
Список учеников	
<input type="text" value="Синичкина Лена"/>	
<input type="text" value="Сидорова Оля"/>	
<input type="text" value="Кузина Катя"/>	
Математический	<input type="text" value="Петрова Лариса Николаевна"/>
Список учеников	
<input type="text" value="Кошкин Вася"/>	
<input type="text" value="Лева Мухин"/>	
<input type="text" value="Тарасенко Гриша"/>	

Рис. 190. Окончательный вид отчета «Ученики по кружкам»

Список «Все объекты Access» в левой части окна базы данных пополнился еще одним отчетом (рис. 191а).

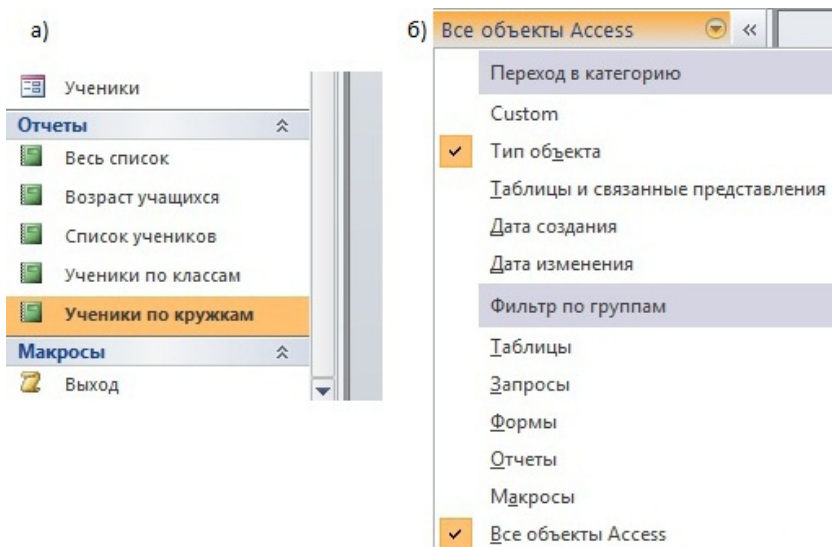


Рис. 191. Все объекты Access

Читатель, наверно, уже обратил внимание на то, что все чаще для открытия нужного объекта приходится пользоваться полосой прокрутки. Если это в тягость, можно открыть меню справа от заголовка списка объектов и установить подходящий фильтр (рис. 191б). Например, чтобы перед вами был только список отчетов.

§ 4.5. Отчет «Количество учеников»

140 ⇔ 157 Следующий отчет будет содержать количество учеников по классам и кружкам. Причем нас будут интересовать только ученики 9–11-х классов. На самом деле последнее будет уже не совсем количеством учеников, поскольку один ученик может заниматься в нескольких кружках. Правильнее говорить о количестве пар «ученик – кружок». Для начала создадим один вспомогательный запрос: «Создание», «Мастер запросов», «Простой запрос», «ОК». Выберем таблицу «Ученики», а в ней поля «ID_ученик» и «Класс» (рис. 192). Не закрывая окна,

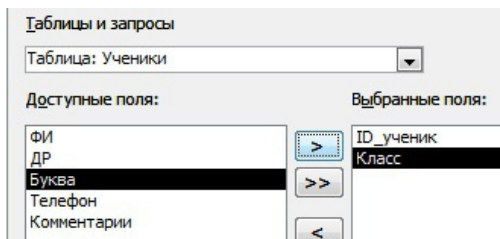


Рис. 192. Выбор полей из таблицы «Ученики»

снова развернем меню «Таблицы и запросы» и выберем таблицу «Ученик_кружок», а из нее только одно поле «Кружок» (рис. 193). Как видно на рисунке, всего мы выбрали три поля из двух таблиц, связанных по полю «ID_ученик». «Далее». Оставим переключатель в положении

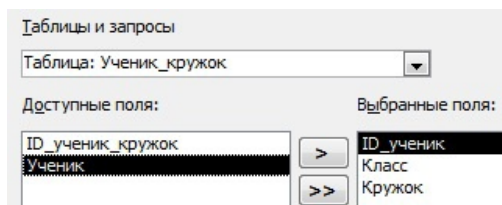


Рис. 193. Выбор полей из таблицы «Ученик_кружок»

«подробный» (рис. 194). «Далее». Сохраним запрос под

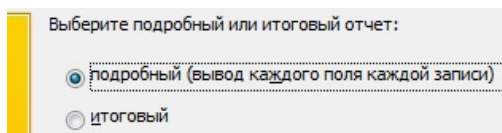


Рис. 194. Все объекты Access

именем «Количество учеников_0». На экране появится табличное представление запроса (рис. 195). Здесь фигури-

ID_ученик	Класс	Кружок
1	9	Математический
3	10	Математический
2	10	Математический
11	9	Вышивание
15	11	Вышивание

Рис. 195. Запрос «Количество учеников_0»

руют только коды учеников. Как мы ранее отмечали, коды уникальны, а имена не всегда. Переведем запрос в режим конструктора (рис. 196). В столбце поля «Класс» в строке

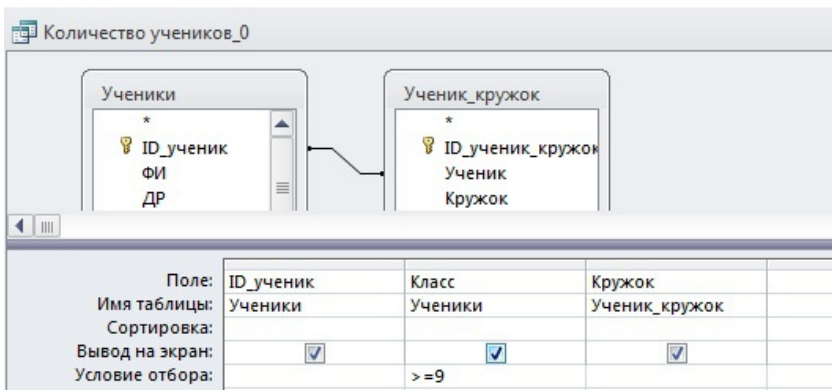


Рис. 196. Запрос «Количество учеников_0» в режиме конструктора

«Условия отбора» пишем: ≥ 9 . Это значит, в запросе будут представлены только записи, относящиеся к ученикам 9-го и выше классов. Закроем конструктор. Убедимся в правильной работе запроса. Перейдем к основному запросу: «Создание», «Мастер запросов», «Перекрестный запрос», «ОК». Выберем источник данных. Для этого установим переключатель в положение «Запросы» и выберем запрос «Количество учеников_0» (рис. 197). «Далее». Выберем поле «Кружок», значения которого будут заголовками строк (рис. 198). «Далее». В качестве заголовков столбцов выберем значения поля «Класс» (рис. 199). «Далее». Осталось поле «ID_ученик». Ему мы поставим в соответствие функцию «Число», как показано на рис. 200. Это значит, что в ячейке, строка которой

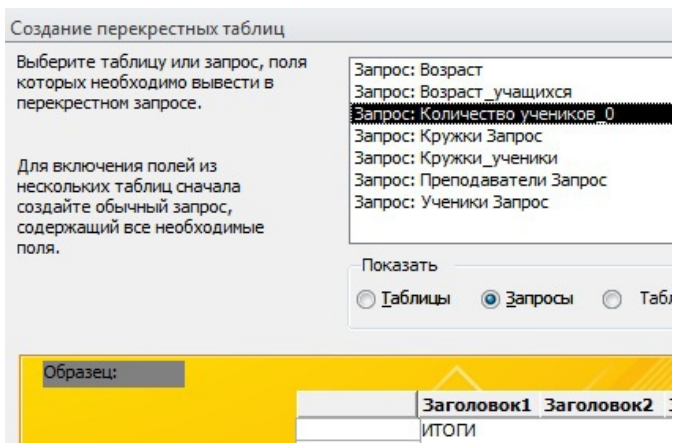


Рис. 197. Выбор запроса

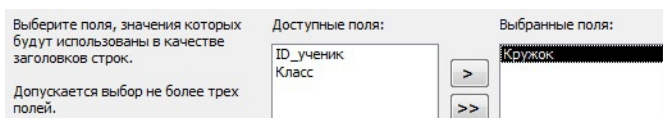


Рис. 198. Определение строк

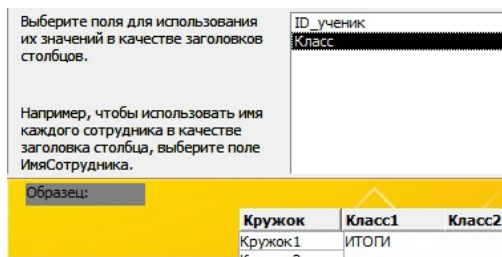


Рис. 199. Определение столбцов

соответствует кружку, а столбец – классу, будет находиться соответствующее число учеников. Сохраним запрос под

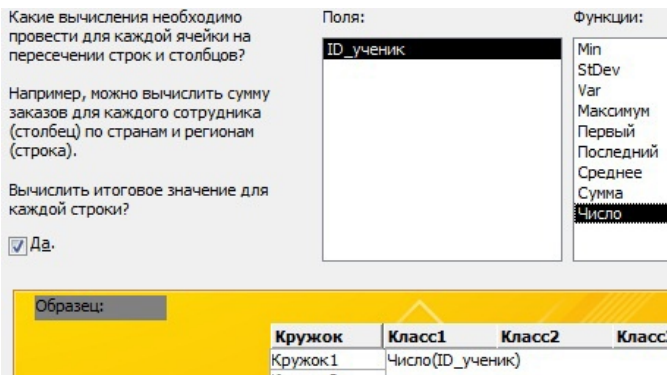


Рис. 200. Выбор функции

именем «Количество учеников». «Готово». На экране появится табличное представление запроса (рис. 201). Пе-

Количество учеников				
Кружок	Итоговое значение	9	10	11
Математический	4	1	3	
Физический	4	2	2	
Моделирования	3	2		1
Баскетбольный	4	2	1	1
Химический	4	1	2	1
Умелые руки	4	1	1	2
Вышивание	3	1	1	1

Рис. 201. Запрос «Количество учеников»

реведем его в режим конструктора и в последнем столбце немного сократим имя вычисляемого поля. В первую строку этого столбца до двоеточия вместо «Итоговое значение» введем текст «Итого» (рис. 202). Во второй строке мы видим одно и то же для всех четырех столбцов

	ID_ученик			
	Класс			
	Кружок			
Поле:	Кружок	Класс	ID_ученик	Итого: ID_ученик
Имя таблицы:	Количество ученико	Количество ученико	Количество ученико	Количество учеников_0
Групповая операция:	Группировка	Группировка	Count	Count
Перекрестная таблица:	Заголовки строк	Заголовки столбцов	Значение	Заголовки строк
Сортировка:				

Рис. 202. Запрос «Количество учеников» в режиме конструктора

«Имя таблицы» – запрос «Количество учеников_0». В третьей строке столбцов «Кружок» и «Класс» стоит отметка «Группировка». Значит, по значениям этих полей производится группировка, а в столбце «ID_ученик» указана уже знакомая нам групповая операция «Count» – число записей. Последнее поле служит для нахождения итогового, т. е. суммарного значения количества записей по каждой строке. Перейдем к созданию отчета. «Создание». «Мастер отчетов». Источник данных – запрос «Количество учеников». Выбранные поля – все, т. е. «Кружок», «Итого», «9», «10» и «11» (рис. 203). Это кажется непривычно, но «9», «10» и «11» здесь – это имена полей. «Далее». Уровни группировки пропустим (рис. 204). «Далее». Сохраним отчет под именем «Количество учеников» (рис. 205). На экране появится созданный мастером отчет (рис. 206). Внимательно рассмотрим его, чтобы сразу оценить, что здесь нуждается в доработке. Перейдем в режим

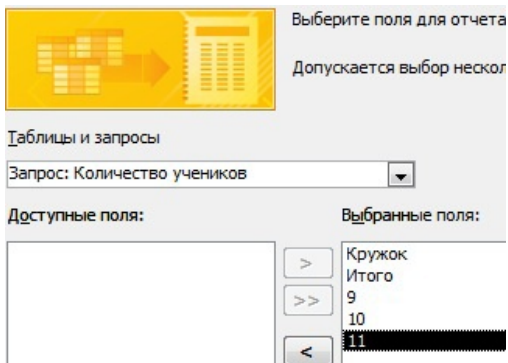


Рис. 203. Выбор источника данных и полей

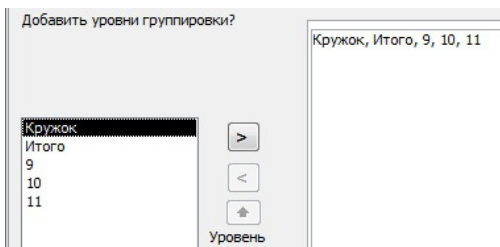


Рис. 204. Уровни группировки

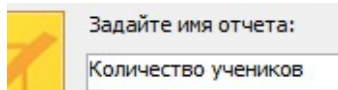
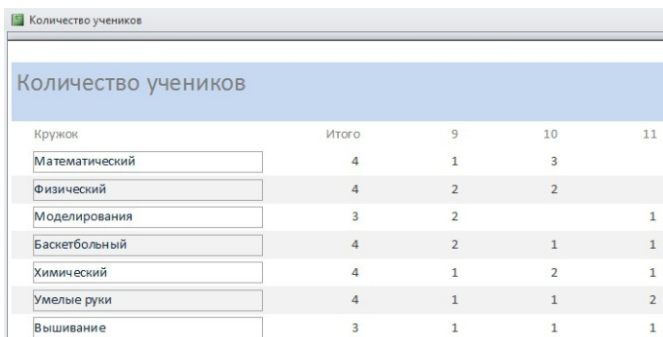


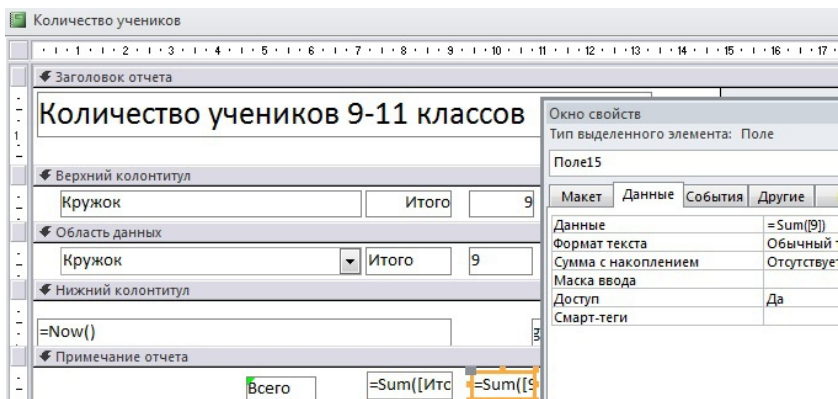
Рис. 205. Сохранение отчета

конструктора (рис. 207). Отредактируем надпись в заголовке отчета. В примечание добавим надпись «Всего» и четыре поля, соответствующие полям из области данных. Эти поля проще просто скопировать, дабы не тратить



Кружок	Итого	9	10	11
Математический	4	1	3	
Физический	4	2	2	
Моделирования	3	2		1
Баскетбольный	4	2	1	1
Химический	4	1	2	1
Умелые руки	4	1	1	2
Вышивание	3	1	1	1

Рис. 206. Отчет «Количество учеников»



Количество учеников 9-11 классов

Заголовок отчета

Верхний колонтитул

Область данных

Нижний колонтитул

Примечание отчета

Свойства: Поле15

Макет | Данные | События | Другие

Данные: =Sum([9])

Формат текста: Обычный

Сумма с накоплением: Отсутствует

Маска ввода:

Доступ: Да

Смарт-теги:

Рис. 207. Отчет «Количество учеников» в режиме конструктора

время на форматирование. Поля будут содержать суммарные значения по столбцам: «=Sum([Итого])», «=Sum([9])», «=Sum([10])» и «=Sum([11])». Для их ввода лучше воспользоваться окном свойств (вкладка «Данные»), как показано на рис. 207. Вероятно, придется немного

отредактировать размеры полей и их расположение. Фон сделаем белым, а цвет шрифта – черным. Пожалуй все! Закроем конструктор с сохранением. Теперь при открытии отчет будет иметь вид, как на рис. 208.

The screenshot shows a window titled "Количество учеников" with a report titled "Количество учеников 9-11 классов". The report contains a table with the following data:

Кружок	Итого	9	10	11
Математический	4	1	3	
Физический	4	2	2	
Моделирования	3	2		1
Баскетбольный	4	2	1	1
Химический	4	1	2	1
Умелые руки	4	1	1	2
Вышивание	3	1	1	1
Всего	26	10	10	6

At the bottom left of the report, it says "28 октября 2019 г." and at the bottom right, "Стр. 1 из 1".

Рис. 208. Окончательный вид отчета «Количество учеников»

Всего, не считая подчиненного, в этой главе мы создали пять различных отчетов, рассмотрев наиболее распространенные вопросы, возникающие при их создании. Во многом возможности отчетов определяются набором тех основных сущностей, которые мы выделили в начале книги, и, разумеется, их свойствами. Так, если мы не отметили такое свойство ученика, как «Пол», то лишили себя,

а значит и пользователя приложения, возможности получать отчеты о количестве мальчиков и девочек, посещающих кружки.

§ 4.6. Редактирование кнопочной формы

148 ⇔ 160 Осталось обеспечить навигацию по созданным отчетам. Ранее мы зарезервировали для отчетов страничку кнопочной формы. Щелкнем по пиктограмме «Диспетчер кнопочных форм» в левом верхнем углу окна базы данных. На экране всплывет окно диспетчера. Выберем «Отчеты» и нажмем «Изменить» или просто дважды щелкнем по соответствующей строке. Появится страничка «Отчеты» (рис. 209). «Создать». В диалоговом окне

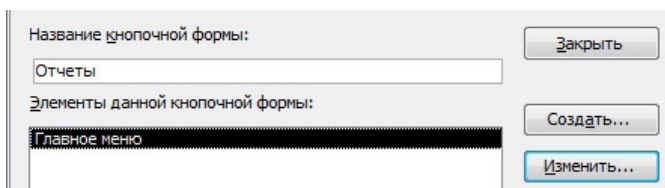


Рис. 209. Окно диспетчера кнопочных форм

в строке «Текст» наберем «Весь список»; в строке «Команда» откроем меню и выберем «Открыть отчет»; в строке «Отчет» в меню выберем «Весь список» (рис. 210). У нас текст «Весь список», который в дальнейшем будет

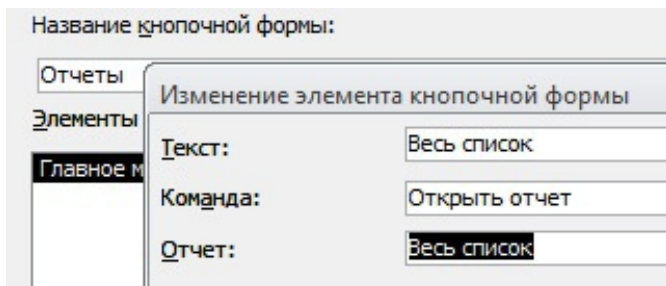


Рис. 210. Параметры элемента кнопочной формы

служить надписью к соответствующей кнопке, совпадает с названием отчета, который должен открываться при нажатии на эту кнопку. В общем случае это не обязательно. Нажимаем «ОК». Далее точно так же создаем кнопки для остальных отчетов: «Возраст учащихся», «Количество учеников», «Ученики по классам» и «Ученики по кружкам». Кнопка «Главное меню», которая служит для возврата на главную страницу кнопочной формы, оказалась сверху. Установим на соответствующей строке указатель и, пользуясь кнопкой «Вниз», переместим ее в конец списка (рис. 211). Нажмем «Заккрыть» и еще раз «Заккрыть».

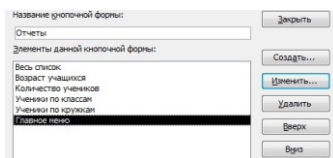


Рис. 211. Страница кнопочной формы «Отчеты»

Теперь откроем кнопочную форму. Нажмем кнопку «Отчеты». На экране появится окно с кнопками для вызова отчетов (рис. 212). Убедимся, что кнопки вызывают

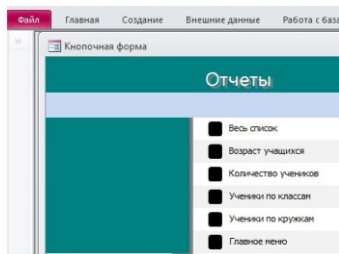


Рис. 212. Окно кнопочной формы «Отчеты»

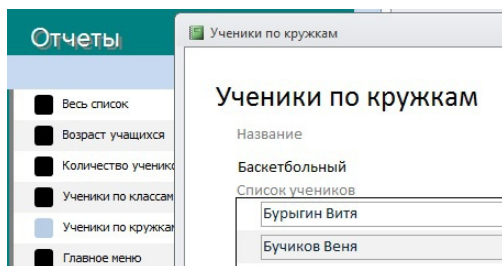


Рис. 213. Вызов отчета

отчеты, соответствующие надписям. Например, при нажатии на кнопку с надписью «Ученики по кружкам» открывается одноименный отчет (рис. 213). Открытый отчет мы можем вывести на печать. Для этого достаточно нажать «Файл» в верхнем меню окна базы данных и в открывшемся вертикальном меню выбрать строку «Печать».

Отладка приложения

157 ⇔ 168 Когда кажется, что дело сделано, начинается самый серьезный этап работы – отладка. Хотя программист в процессе разработки приложения занимается отладкой постоянно, только сейчас можно сконцентрировать все внимание на вопросах надежности приложения. Для этого еще раз проверим в работе все его компоненты. В частности, заметим, что набор данных, с которым мы до сих пор работали, возможно, не отражает все ситуации, которые могут возникнуть при эксплуатации приложения. Кроме того, мы ни разу не удаляли первичные данные. Откроем кнопочную форму, перейдем на ее страничку «Первичные данные» (рис. 110 на с. 100) и откроем форму «Список учеников». Допустим, один из учеников, например Петров Коля, переехал в другой город. Удалим запись, в которой поле «ФИ» имеет значение «Петров Коля». Для этого щелкнем левой клавишей мыши по области выделения в соответствующей строке. Область выделения окрасится в черный цвет. Нажмем на клавиатуре компьютера «Delete». Появится диалоговое окно с просьбой подтвердить удаление. После подтверждения в ленточной форме пропадет соответствующая строка.

Закроем форму и откроем «Список кружков». Сделаем двойной щелчок по названию кружка «Физический». Всплывет форма с информацией о кружке, в которой представлен список членов кружка. Как мы видим (рис. 214), на месте записи «Петров Коля» появилась пустая строка. При этом в таблице «Ученик_кружок» по-прежнему



Рис. 214. Физический кружок

хранится код Петрова Коли, но к этому коду отсутствует подстановка. Закроем форму и откроем «Список кружков». Удалим в нем кружок «Умелые руки». Если теперь просмотреть отчеты, покажется, что вся информация о деятельности кружка бесследно исчезла. Но не тут-то было! Закроем кнопочную форму и откроем таблицу «Ученик_кружок». Ее фрагмент представлен на рис. 215. Мы видим ряд записей, в которых отсутствует название кружка. Опять та же история: запись о связи между учеником и кружком сохранилась, код кружка сохранился, но отсутствует подстановка к коду, поскольку удалена запись в таблице «Кружки», которая связывала код и название.

13	Попова Ира	Моделирования	
14	Кузина Катя	Математический	
15	Иванова Даша		
16	Карпов С аша		
17			
18	Потапов Федя		
19	Бучиков Веня	Физический	
20	Тарасенко Гриша	Физический	
21		Физический	
22	Перепелкина Оля	Физический	
23	Рысакав Рома	Химический	

Рис. 215. Таблица «Ученик_кружок»

В двух записях – пустое значение «ФИ». Здесь был «Коля Петров». Его код сохранился, но подстановка пропала. Таким образом, имеет место нарушение целостности данных. В таблице «Ученик_кружок» присутствуют неактуальные записи. Этот недостаток мы должны устранить. Пришло время открыть маленький секрет. Прежде чем демонстрировать удаление ученика и кружка, автор сделал копию файла «Кружки.accdb» и все эти манипуляции выполнил на копии. Как говорится, «в процессе съемок никто не пострадал». Жалко ведь трудов по вводу данных. Удалим копию базы данных и сделаем новую копию

исходного файла. Откроем ее. В кнопочной форме перейдем на страничку «Первичные данные». Откроем форму «Список преподавателей» и удалим запись «Крылов Николай Кузьмич», руководитель кружка «Умелые руки». Закроем форму и откроем «Список кружков». На месте руководителя кружка «Умелые руки» мы видим пустую строку (рис. 216). Уход руководителя не всегда означает

Баскетбольный	Солдатов Петр Иванович
Вышивание	Попова Светлана Борисовна
Математический	Петрова Лариса Николаевна
Моделирования	Силин Иван Петрович
Умелые руки	
Физический	Силин Иван Петрович
Химический	Фролова Тамара Николаевна

Рис. 216. Список кружков

прекращение работы кружка. Мы можем открыть меню и назначить нового руководителя из списка преподавателей. Здесь угрозы целостности данных нет. Выйдем из приложения. «Испорченную» копию можно удалить. Откроем исходный файл. Закроем кнопочную форму, щелкнув по крестику в правом верхнем углу ее окна. Выберем пункт меню «Работа с базами данных». Щелкнем по пиктограмме «Схема данных». В открывшемся окне

схемы данных (рис. 27 на с. 35) дважды щелкнем по линии, соединяющей поле «ID_ученик» таблицы «Ученики» с полем «Ученик» таблицы «Ученик_кружок». Откроется окно с заголовком «Изменение связей» (рис. 217а). Установим флажки: «Обеспечение целостности данных»,

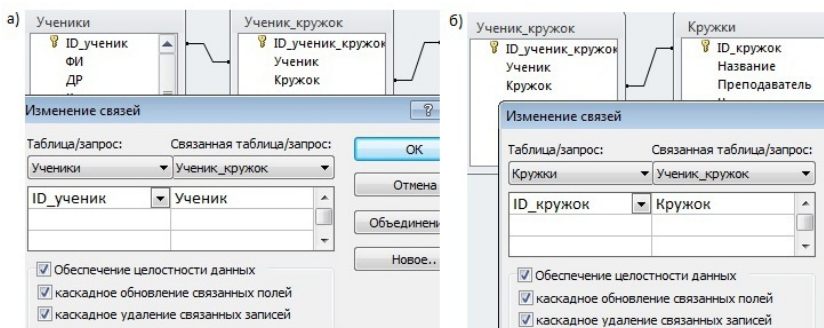


Рис. 217. Изменение связей

«Каскадное обновление связанных полей» и «Каскадное удаление связанных записей». «ОК». Аналогично дважды щелкнем по линии, соединяющей поле «Кружок» таблицы «Ученик_кружок» с полем «ID_кружок» таблицы «Кружки», и установим такие же флажки (рис. 217б). У концов линий появились значки: «1 – ∞» и «∞ – 1» соответственно (рис. 218). Это значит, что каждый ученик может заниматься в нескольких кружках, а в каждом кружке может заниматься несколько учеников. Теперь система берет на себя заботу об обеспечении целостности данных.

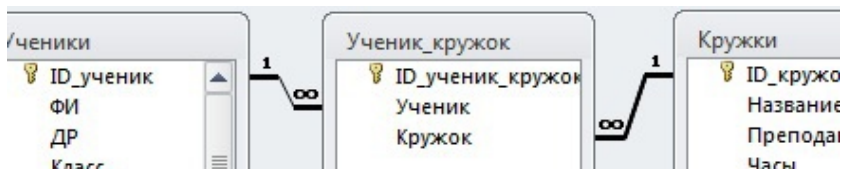


Рис. 218. Схема данных

В случае удаления записи об ученике будут удаляться все соответствующие ему записи из связанной таблицы «Ученик_кружок». Если удалили запись о Коле Петрове, то автоматически система удалит все записи из таблицы «Ученик_кружок», в которых фигурирует Коля Петров, точнее его код. В случае удаления записи о кружке будут удаляться все соответствующие записи таблицы «Ученик_кружок».

Зададимся вопросом: что происходит при удалении записей? Можете легко проверить, что при удалении записей размер файла не уменьшается. Говорят, что записи удаляются логически, но не физически. В процессе разработки приложения мы часто вносим исправления в созданные нами объекты базы данных. Но даже если исправление состоит в том, что мы удалили какой-либо объект, размер файла не уменьшится. Таким образом, как в процессе разработки приложения, так и в процессе его эксплуатации размер файла будет только увеличиваться.

Если нам придется часто добавлять и удалять записи, значительная часть занимаемой приложением памяти будет хранить «мусор». Нерациональное использование памяти само по себе плохо, но это также ведет к замедлению работы приложения. Чтобы устранить эту проблему, применяют процедуру сжатия файла. Выберем пункт верхнего меню «Файл» и щелкнем по большой кнопке с надписью «Сжать и восстановить базу данных» (рис. 219). В нашем

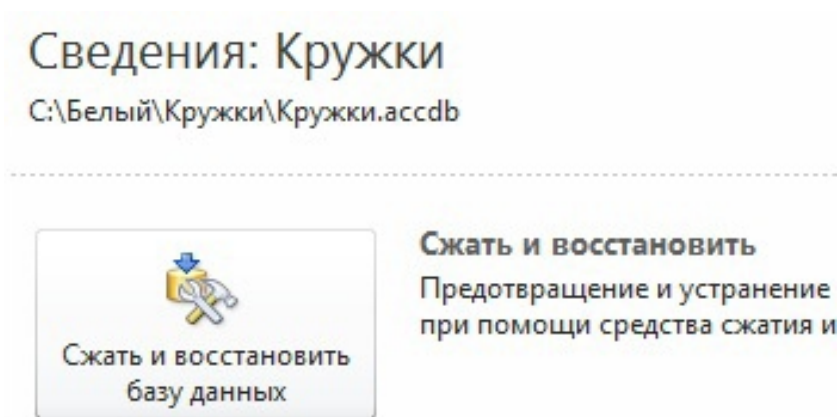


Рис. 219. Сжатие базы данных

случае к концу разработки приложения его файл занимал 1,38МБ, после сжатия – 0,99МБ. Однако дабы не нагружать пользователя лишними заботами, лучше сделать так, чтобы сжатие время от времени происходило автоматически. Для этого предусмотрен специальный параметр базы

данных. Выберем: «Файл», «Параметры», «Текущая база данных». В открывшемся окне устанавливаем флажок перед строкой «Сжимать при закрытии» (рис. 220). Это

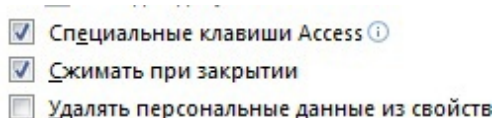


Рис. 220. Схема данных

значит, что сжатие вашей базы данных будет происходить при каждом ее закрытии.

Теперь приложение можно сдать в эксплуатацию. Но для программиста и это еще не конец трудов. Как только заказчик приступит к работе с приложением, его, т. е. заказчика, непременно начнут посещать фантазии, порой достойные внимания Бехтерева. Он будет постоянно тормозить программиста, и бедняге придется создавать все новые и новые поля, запросы, формы, отчеты...

Историческая справка

160 ⇔ 173 Microsoft – крупнейшая транснациональная компания по производству программного обеспечения и аксессуаров для персональных компьютеров. Программы Microsoft переведены более чем на 45 языков.

Основали компанию в 1975 году два друга-студента – Билл Гейтс и Пол Аллен. Гейтсу тогда исполнилось 20 лет,

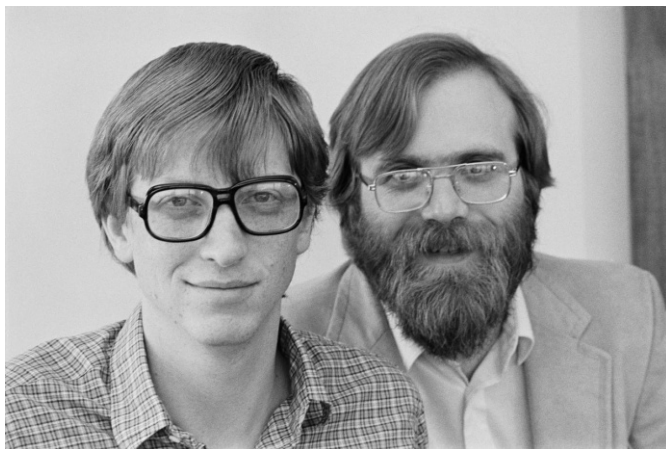


Рис. 221. Билл Гейтс и Пол Аллен

а Аллену 22 года. Другими и единомышленниками они были еще со школьной скамьи в Сиэтле. Восьмиклассник Гейтс и десятиклассник Аллен познакомились на занятиях по программированию. Оба все свободное время посвящали компьютеру, часто в ущерб другим предметам.

При этом Гейтс с выраженным математическим мышлением более тяготел к программированию, а Аллена больше интересовала техническая сторона вычислительной техники. Главный капитал, который могли тогда вложить



Рис. 222. Билл Гейтс



Рис. 223. Пол Аллен

в бизнес друзья, – это их мозги, идеи и желание работать. Проект оказался успешным. Всего через 11 лет

Билл Гейтс, а немного позже и Пол Аллен стали миллиардерами. Возможно, кто-то скажет: «Сбылась американская мечта». Но для Гейтса и Аллена это было только начало пути. В 1983 г. Пол Аллен покидает Microsoft, а в 2008 г. Билл Гейтс оставляет должность руководителя корпорации. В 2010 г. Гейтс оставляет также пост исполнительного директора Microsoft, чтобы полностью посвятить себя благотворительным проектам. Среди состоятельных людей мира Гейтс является абсолютным чемпионом по расходам на благотворительность. С 1994 по 2010 г. он вложил в фонд Билла и Мелинды Гейтс 28 млрд долларов. Гейтс поддерживает проекты, связанные с экологией, поиском чистой и альтернативной энергии. В феврале 2010 г. он выступает с предложением ко всем миллиардерам о передаче половины своих состояний на благотворительность.

Пол Аллен после ухода из Microsoft субсидирует создание первого частного суборбитального корабля SpaceShipOne, на котором были осуществлены два успешных космических полета. В 2005 г. финансирует строительство радиотелескопа для поиска внеземных цивилизаций. В 2011 г. становится одним из основателей компании Stratolaunch Systems, разрабатывающей системы доставки грузов на орбиту. На одной из своих яхт, снабженной двумя

подводными мини-лодками, Аллен проводит подводные археологические экспедиции. К сожалению, в 2018 г. болезнь прервала жизнь этого выдающегося человека.

В настоящее время продукция Microsoft занимает значительную долю рынка программного обеспечения. Наибольшую известность получили операционная система Windows, текстовый процессор Word, табличный процессор Excel, реляционная СУБД Access. На данный момент председателем совета директоров Microsoft является Джон Томпсон, главным управляющим – Сатья Наделла.

Фотографии (рис. 221–223) взяты с сайта:

<https://yandex.ru/images>

Список литературы

- [1] Гурвиц, Г. А. Microsoft Access 2010. Разработка приложений на реальном примере / Г. А. Гурвиц. – Санкт-Петербург : БХВ – Петербург, 2010. – 496 с.
- [2] Дейт, К. Дж. Введение в системы баз данных / К. Дж. Дейт. – Москва : Вильямс, 2001. – 1072 с.
- [3] Бекаревич, Ю. Б. Самоучитель Access 2010 / Ю. Б. Бекаревич. – Санкт-Петербург : БХВ – Петербург, 2011. – 432 с.

**Вышли в печать книги серии
«Математика не для ЕГЭ»:**

Белый Е. К. Символы и их творцы. – Петрозаводск: Издательство ПетрГУ, 2018. – 72 с.

Белый Е. К. Вредная геометрия. – Петрозаводск: Издательство ПетрГУ, 2017. – 36 с.

Белый Е. К. Египетский счет. – Петрозаводск: Издательство ПетрГУ, 2017. – 36 с.

Белый Е. К. Прогрессии. – Петрозаводск: Издательство ПетрГУ, 2016. – 132 с.

Белый Е. К., Дорофеева Ю. А. Алгебраические уравнения. – Петрозаводск: Издательство ПетрГУ, 2015. – 240 с.

Книги можно найти в сети Интернет, в частности на сайте «Учительский портал»: <https://www.uchportal.ru/>

Учебное электронное издание

Белый Евгений Константинович

Математика не для ЕГЭ

Введение в Microsoft Access

Учебное электронное пособие
для учащихся средних школ

Редактор *Е. Е. Порывакина*

Оригинал-макет, электронная версия *Е. К. Белого*

Оформление обложки *Е. Ю. Тихоновой*

Подписано к изготовлению 25.01.20. 1CD-R. 9.4 Мб.

Тираж 100 экз. Изд. № 236

Федеральное государственное бюджетное
образовательное учреждение высшего образования

ПЕТРОЗАВОДСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ

185910, г. Петрозаводск, пр. Ленина, 33

<https://petsu.ru>

Тел.: (8142) 71-10-01

Изготовлено в Издательстве ПетрГУ

185910, г. Петрозаводск, пр. Ленина, 33

URL: press.petsu.ru/UNIPRESS/UNIPRESS.html

Тел./факс: (8142) 78-15-40

nvpahomova@yandex.ru