

Лабораторная работа № 6 «СОЗДАНИЕ НОВОЙ БАЗЫ ДАННЫХ»

Цель работы

Данная лабораторная работа предназначена для приобретения начальных навыков работы с таблицами реляционной базы данных, а именно:

- а) создавать таблицы, задавая полям различные типы данных,
- б) вводить и редактировать данные в таблицах,
- в) организовывать связи между таблицами.

Создание пустой базы

База данных – это файл специального формата, содержащий информацию, структурированную заданным образом. Все объекты базы данных Access хранятся в одном файле с расширением .mdb. Файл новой БД занимает около 96 Кбайт содержит скрытые системные таблицы, в которые в дальнейшем будет заноситься информация обо всех других объектах БД. По мере создания новых объектов и заполнения самой БД размер этого файла будет расти, однако на протяжении всего периода существования БД вся эта совокупность элементов Access рассматривается как один физический файл.

С точки зрения пользователя, создание пустой базы данных – это некая формальная процедура, извещающая систему о ваших намерениях в дальнейшем заполнить этот файл реальными таблицами, запросами и другими объектами.

Создание объектов можно выполнять с помощью *Мастера* – специальной программы, ускоряющей создание структуры объекта, – или вручную. Для лучшего понимания структуры базы данных мы, в основном, будем ориентироваться на работу вручную – в режиме Конструктора (рис. 3).

ЗАДАНИЕ 1. Создание таблиц в режиме Конструктора

1. Запустите программу Access.
2. В появившемся диалоговом окне, установите флажок напротив опции новая база данных и подтвердите операцию.
3. В следующем диалоговом окне вы должны выбрать папку для базы данных и задать ее имя (*Факультет*).

На экране появится окно с шестью вкладками, это и есть ваша база данных (она пока пустая).

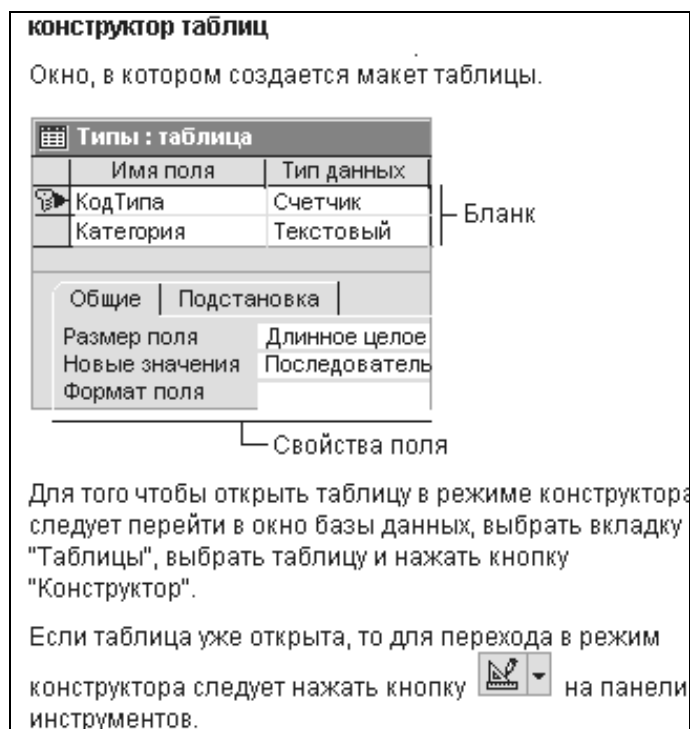



Рис. 3. Инструкция для открытия и макет таблицы в режиме Конструктора

4. Перейдите на вкладку Таблицы.

5. Щелкните мышью по кнопке Создание таблицы в режиме конструктора

Основываясь на спроектированных нами таблицах (табл. 3 - 7), будем создавать поля со свойствами, описанными в этих таблицах. Начнем с таблицы *Учетная карта*.

6. В столбец Имя поля занесите имена столбцов будущей таблицы (рис.4). Напомним, что при этом нельзя использовать некоторые символы, в том числе пробелы, точки и запятые. В столбце Тип данных выберите (используя кнопку вызова списка ) тип данных. А то, что заносится в столбец Описание затем появляется, в виде комментариев, в строке состояния.

7. Названия полей даны латинскими буквами. Чтобы из-за этого не возникало неудобств при работе с таблицей, каждому полю задайте подпись

на русском языке. Для этого надо перейти на вкладку Свойства поля и задать в строке Подпись сокращенное название поля на русском языке.

Свойство Подпись – это строковое выражение длиной до 2048 символов. Оно определяет тот текст, который будет выводиться в заголовке столбца в режиме таблицы. По умолчанию в качестве заголовка выводится имя поля. В подписи можно использовать любые символы, в том числе пробелы и точки.

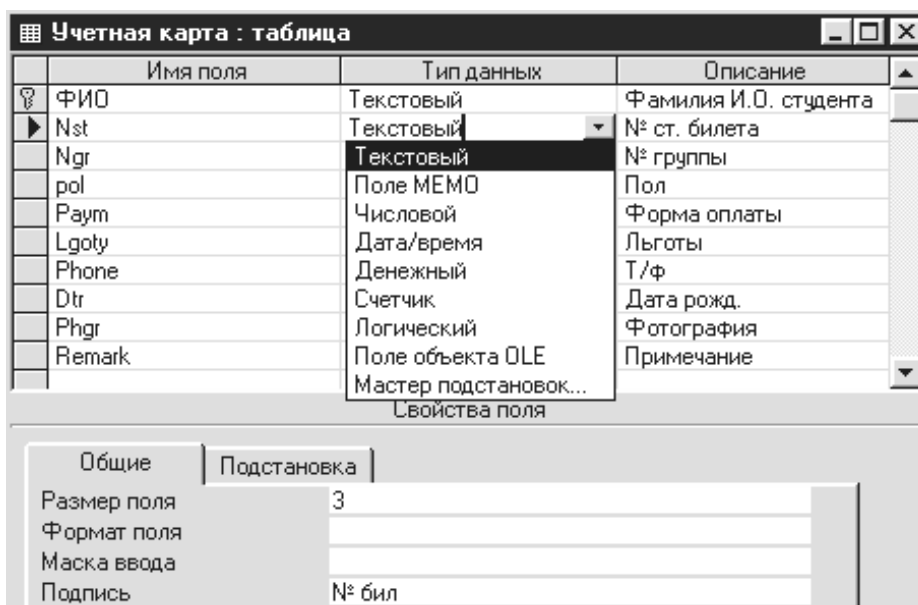


Рис. 4. Название и типы полей таблицы *Учетная карта*.

Показан разворачивающийся список типов полей по кнопке вызова

8. Здесь же, в строке Размер поля, на основе табл. 3 задать размер для числовых и текстовых полей.

9. После ввода полей и типов данных желательно задать ключевое поле. Для этого необходимо щелкнуть правой клавишей мыши по полю *ФИО* и, в появившемся меню, выполнить команду Ключевое поле.

11. После ввода всех полей, их типов, размеров и названий закройте окно Конструктора и, при запросе о сохранении, задайте имя *Учетная карта*.

12. Аналогично создайте еще четыре таблицы, структура которых показана на рис. 5. Длину полей, подписи к ним и другие свойства полей выбирайте в согласии с таблицами 4 – 7. Имена полей, их типы и назначения показаны на рисунке 5.

13. Закройте таблицы с сохранением, дав им соответствующие имена.

14. Сохраните данную базу данных на диске или дискете для последующей работы. Для этого необходимо закрыть ACCESS и скопировать файл базы одним из средств Windows.

Изменение типа полей и реорганизация таблиц

В режиме конструктора все изменения, касающиеся структуры объектов, необходимо сохранять. Изменения, касающиеся структуры таблиц, желательно выполнять на этапе проектирования, еще до ввода данных, когда еще не созданы формы и запросы. В противном случае изменение типов полей может повлечь за собой потерю данных, о чем Access предупреждает при попытке сохранить изменения.

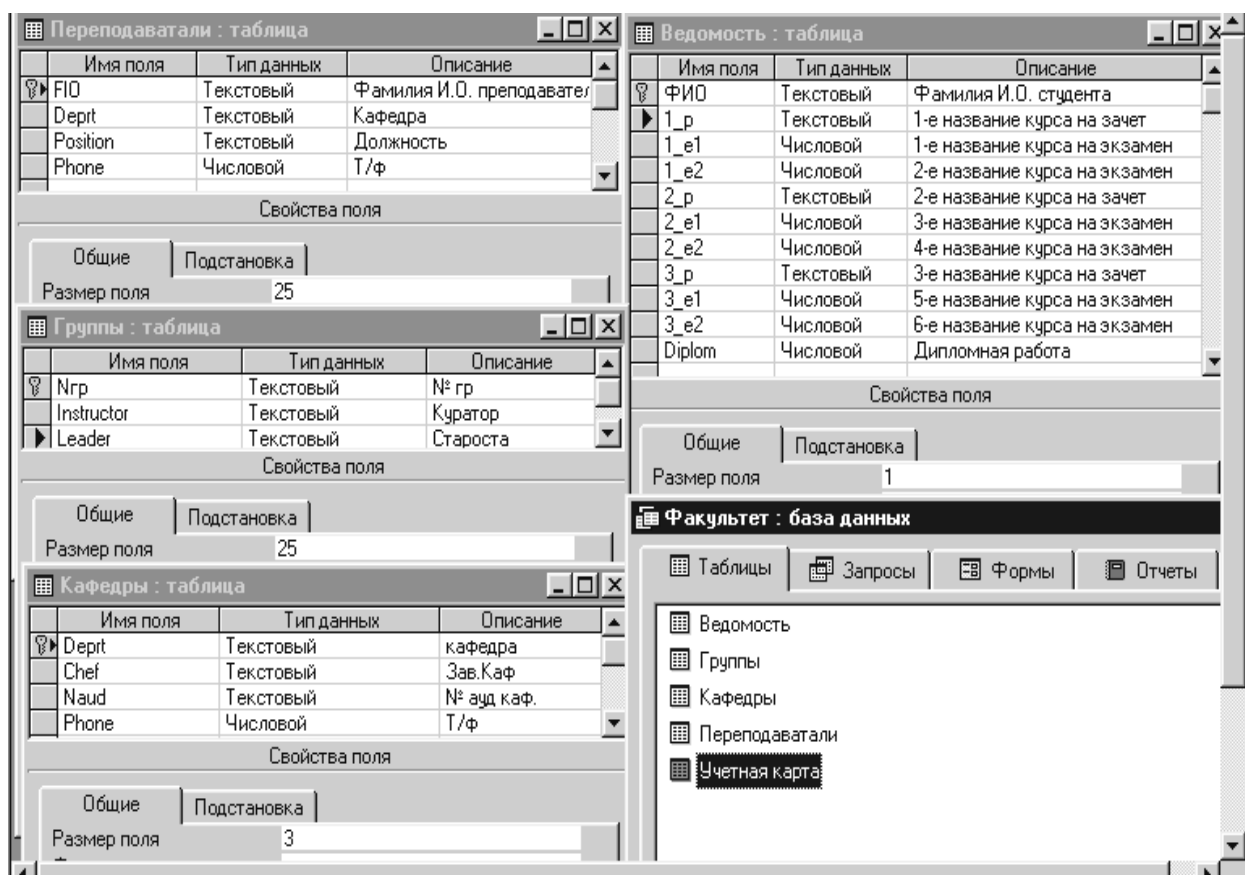


Рис. 5. База данных *Факультет* и структура ее четырех таблиц

- Чтобы *изменить имя поля* таблицы, нужно дважды щелкнуть на текущем имени поля и ввести новое имя.
- Чтобы *изменить тип поля данных* для уже существующего поля, надо щелкнуть на кнопке раскрывающегося списка в столбце Тип Данных и выбрать новый тип данных.

- Для добавления или удаления полей можно воспользоваться контекстным меню, установив курсор на требуемой строке в списке полей, и выбрать соответствующую команду.

- При *копировании полей* копируется только их структура, а не сами значения. Для этой процедуры можно использовать средства Windows для работы с буфером обмена.

- Для удаления поля так же используются традиционные средства Windows (буфер обмена или клавиша <Delete> на выделенном поле).

Обратим внимание на одну особенность всех баз данных. Таблица баз данных не является самостоятельным документом. Сама база – это документ. Ей соответствует файл на диске, можно сделать его копию. Структура таблиц входит в состав общего файла базы данных наряду с запросами, формами и другими объектами. При изменении структуры таблицы СУБД всегда выдает запрос на сохранение изменений.

Но содержание таблиц – это совсем другое дело. Его нельзя сохранить принудительной командой или, наоборот, отказаться от его сохранения. Все изменения в таблицах сохраняются автоматически *в режиме реального времени*. Режим реального времени означает, что, пока мы работаем с таблицей, происходит ее непрерывное сохранение. Как только заканчивается ввод данных в одно поле и происходит переход к следующему полю, данные немедленно записываются на жесткий диск.

Экспериментируя с таблицами, надо знать, что все изменения, которые вносятся в их содержание, имеют необратимый характер. Нельзя что-то изменить, удалить, а потом отказаться от сохранения и вернуться к исходному варианту.

Ввод и редактирование данных в таблице

Когда структура таблицы создана и откорректирована, можно приступать ко вводу данных. Начнем заполнение с таблицы *Учетная карта*. Чтобы добавить в таблицу первую запись, надо выполнить следующие действия.

1. Двойным щелчком (или с помощью кнопки *Открыть*) откройте таблицу – она будет представлена в режиме таблицы, а указатель будет помещен в первое поле первой записи.

2. Щелкнуть в первом поле пустой записи и ввести значение для данного поля. Как только в текущую строку будут вводиться данные, в таблице появится новая пустая запись. (Если первым в таблице стоит поле с типом *Счетчик*, то оно заполняется автоматически). Для перехода в новое поле можно пользоваться клавишами <Tab>, (<Enter> или <→>).

3. Процедуру ввода повторять до тех пор, пока не будут введены данные для всех полей таблицы. Когда введено последнее поле и нажата клавиша <Tab>, (<Enter> или <→>), точка вставки переместится в первое поле новой записи.

Если понадобится внести изменения в заполненную таблицу: отредактировать данные в полях, удалить или добавить записи, то можно пользоваться всеми приемами редактирования Windows: вставка и замена символов в указанном месте поля, работа с буфером обмена и командами горизонтального и контекстного меню. Прием *протягивание* можно выделять группу записей или несколько столбцов (для копирования или удаления), можно выделять часть таблицы и переносить выделенные фрагменты через буфер обмена в рабочие листы Microsoft Excel или таблицы Microsoft Word.

Данные для ввода в таблицу приведены на рис. 6. Для того, чтобы *уменьшить или увеличить ширину столбца*, можно использовать стандартный прием Windows – *перемещение*: установить курсор на границе столбца, зафиксированной левой кнопкой мыши переместить ее в требуемую сторону.

ЗАДАНИЕ 2. Ввод и редактирование данных в полях таблицы

1. Откройте таблицу *Учетная карта*. и введите в таблицу первые две-три записи (рис. 6). Обратите внимание новые записи в Access «добавляются» только внизу таблицы, в так называемую *временную новую запись*. Разместить новую запись между уже существующими или в начало таблицы – нельзя. (Чтобы визуально расположить записи в другом порядке, например, по алфавиту, можно воспользоваться функцией *Сортировка* из пункта горизонтального меню *Записи*).

2. Обратите внимание, что новая запись отмечена «звездочкой» (✳). Но когда указатель перемещается в первое поле новой записи, «звездочка»

изменится на значок текущей записи (▶). Когда в поле новой записи вводятся данные, указатель меняет свой вид на пишущий карандаш ✎. Рис. 6 иллюстрирует именно этот момент.

3. Обратите внимание, что в логическое поле *Льготы* значение «Да» не надо вводить: достаточно щелкнуть в заготовленной клетке левой кнопкой мыши – появится галочка (☑) вместо «Да».




4. Обратите внимание, что для числовых полей (*Т/ф*) можно вводить только цифры: при попытке ввести другие символы Access выдает предупреждение о несовпадении типа поля и данных.

Учетная карта : таблица										
	ФИО	№ бил	№ гр	Пол	Форма опл.	Льготы	Т/ф	Дата рожд.	Ф/гр	
	Андреева О.С.	1-01	А-1	ж	Б	<input type="checkbox"/>	561483	12.07.83		золотая мед
	Антонова Т.И.	1-04	Б-1	ж	К	<input type="checkbox"/>	326454	07.11.83		победитель р
	Арсентьев В.Р.	1-02	А-1	м	Б	<input checked="" type="checkbox"/>		05.10.82		многодетная
	Безногов С.Ю.	2-12	Б-2	м	К	<input type="checkbox"/>	865425682	21.09.84		
	Белов С.П.	1-03	А-1	м	К	<input type="checkbox"/>	710850	29.01.82		
	Бережной П.Р.	2-07	А-2	м	К	<input type="checkbox"/>	448532	08.01.83		
	Верхов А.В.	1-05	Б-1	м	К	<input checked="" type="checkbox"/>	357849	25.03.83		Служба в Чеч
	Галкин И.Д.	2-08	А-2	м	К	<input type="checkbox"/>	438162	04.04.84		
	Денисов И.О.	1-06	Б-1	м	К	<input type="checkbox"/>	743285	20.02.82		
	Драгунов А.А.	2-09	Б-2	м	Б	<input type="checkbox"/>	824768924	24.05.84		
	Елисеева И.В.	2-10	Б-2	ж	Б	<input checked="" type="checkbox"/>	643091	08.10.83		Сирота
	Леонтьев М.В.	2-14	А-2	М	Б	<input checked="" type="checkbox"/>	557829	25.01.85		Служба в Чеч
	Романова С.Л.	2-12	Б-2	ж	Б	<input checked="" type="checkbox"/>	945751	31.08.83		
	Семенов А.Л.	2-15	А-2	м	Б	<input type="checkbox"/>	570860	23.12.83		
	Торова Р.К.	2-16	Б-1	ж	Б	<input checked="" type="checkbox"/>	348064	02.03.80		многодетная
✎	Як.	2-11	Б-2	м	Б	<input type="checkbox"/>		25.03.84		
*						<input type="checkbox"/>				

Рис. 6. Таблица *Учетная карта* открыта в режиме таблицы для ввода записей. При вводе новой записи значок изменяется на изображение пишущего карандаша, а внизу таблицы появляется новая временная запись

5. Для поля *Дата рожд.* по умолчанию предусмотрен краткий формат даты **дд.мм.гг.** Если в качестве разделителя использовать запятую, то Access автоматически преобразует ее в точку.

6. Для поля типа OLE (*Ф/гр*) оставьте пустое место. Просмотр полей типа OLE возможен только в режиме *Формы*. При изучении работы с *Формами* рассмотрим, как внедрять в таблицу поля такого типа.

7. В поле *Примечание* можно вводить любую текстовую информацию.
8. Исправьте в первой записи фамилию **Андреева О.С.** на **Андропова О.С.** Перейдите в следующее поле.
9. Отмените сделанное изменение, используя кнопку *Отменить*  на панели инструментов или клавиши Ctrl+Z.
10. После ввода 2 - 3 записей из таблицы рис. 6 введите в новую запись данные о себе.
11. Выделите только что введенную запись (щелкнув на кнопке выбора записи или поместив курсор в любое место записи, вызвать из горизонтального меню *Правка>Выделить* запись, затем с помощью контекстного или горизонтального меню вырежьте запись в буфер обмена. На предупреждение Access об опасности удаления, ответьте «Да».
12. Вставьте вырезанную запись с помощью команды горизонтального меню *Правка>Добавить из буфера*. Обратите внимание, что обычный способ вставки из буфера с помощью кнопки *Вставка*  в данном случае не работает.
13. Снова выделите эту же запись и удалите ее, нажав клавишу <Delete> на клавиатуре, или выполнив команду *Правка>Удалить запись*.
14. Закройте таблицу, используя кнопку *Закреть*  или ту же команду горизонтального меню.

Форматирование данных и контроль ввода информации

При заполнении таблиц БД очень важно обеспечить достоверность и корректность данных в таблицах, а также представить их в максимально удобном для эффективной работы виде. Ключевая роль в решении этой задачи принадлежит таким свойствам полей, как *Формат поля*, *Маска ввода* и *Условие на значение*.

Форматирование

Форматирование данных позволяет определить способ отображения на экране вводимых данных. Изменив формат, можно представить данные в более привлекательном, информативном и доступном виде. Форматы применяются строго для определенных типов данных и доступны для любого типа, за исключением типа Поле объекта OLE.

В Access используются две разновидности форматов: *встроенные* и *специальные*. Последние создаются с помощью специальных символов. Для

каждого типа полей существуют свои настройки форматирования и свои специальные символы.

Форматы для текстовых и MEMO- полей создаются с помощью четырех специальных символов (табл. 8).

Таблица 8. Символы форматирования для текстовых и MEMO-полей

Символы	Описание	Примеры форматов (вводимые данные – вид на экране)
1	2	3
@	Прототип символа. В форматлируемой строке вместо @ отображается введенный символ или, если символ не указан, выводится пробел.	@@@-@@-@@@@ (465073799 – 465-07-3799) @@@@@@@@@@@@ (465-07-3799 – 465-07-3799)

Продолжение табл. 22

1	2	3
<	Переводит все символы на нижний регистр.	иванов иванов ИВАНОВ иванов Иванов иванов
>	Переводит все символы на верхний регистр.	иванов ИВАНОВ ИВАНОВ ИВАНОВ Иванов ИВАНОВ

Так, например, для полей, содержащих фамилии и инициалы, во всех таблицах БД *Факультет* можно указать формат > (рис. 7), и тогда при вводе информации можно не следить за регистром: все фамилии и инициалы будут отображаться заглавными буквами.

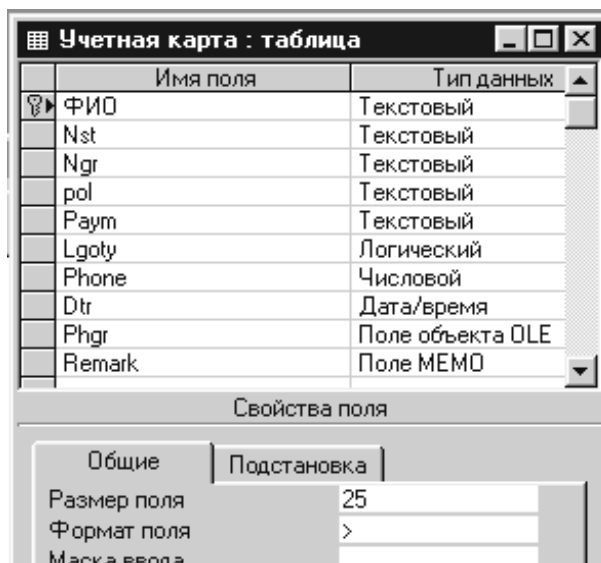


Рис. 7. Для поля ФИО указан формат >, в результате чего данные этого поля выводятся прописными буквами

Для форматирования числовых и денежных полей программа Access предлагает ряд встроенных форматов. Все они включены в список опции Формат поля (рис. 8) в виде пар: название формата и образец его использования.

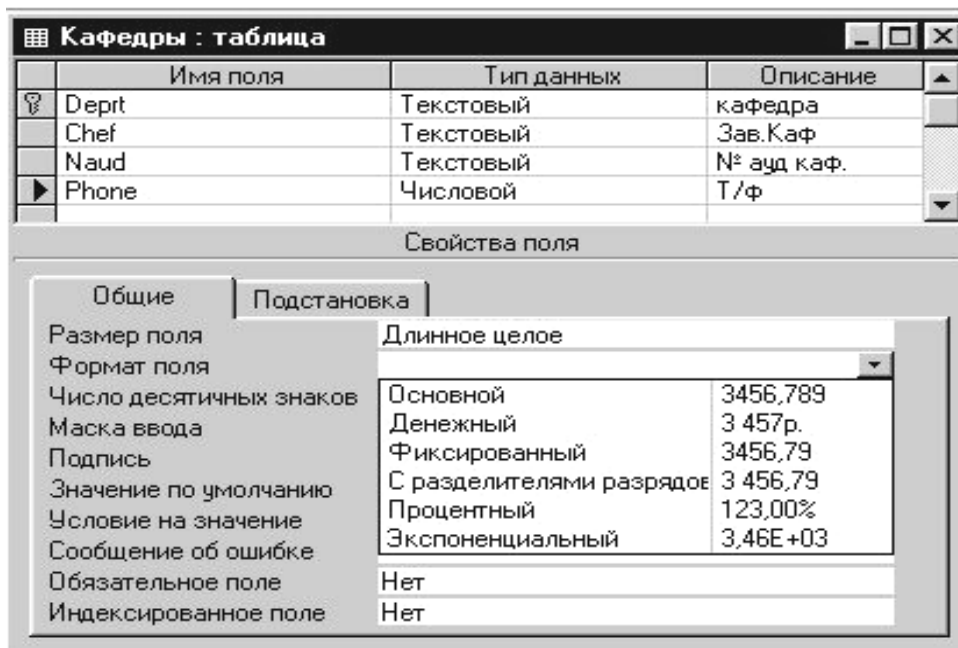


Рис. 8. Список числовых форматов, предлагаемых в окне конструктора для свойства `Формат` поля

Наиболее распространенные *специальные форматы* создаются с помощью символов, представленных в табл. 9.

Таблица 9. Символы, используемые в специальных форматах для числовых и денежных полей

Символ	Описание
.	(точка) Десятичный разделитель.
,	(запятая) Разделитель тысяч.
0	Прототип разряда. Независимо от числа символов 0, целая часть числа будет отображаться полностью. Например, число 345,6 в

	формате 00000,00 выглядит как 00345,60
#	Прототип разряда. Независимо от числа символов #, целая часть числа будет отображаться полностью. Например, число 345,6 в формате #####.## выглядит как 345,6
%	Процентный формат. Значение умножается на 100 и выводится со знаком процентов.
E- или e-	Экспоненциальная нотация с выводом знака минус перед отрицательным показателем и без знака перед положительным показателем. Используется вместе с другими специальными символами, например, 0.00E-00.

Список встроенных форматов для полей типа Дата/время показан на рис. 9.

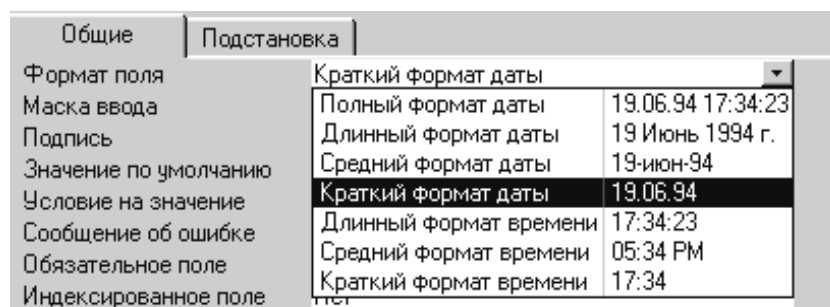


Рис. 9. Список встроенных форматов полей типа Дата/время

Здесь для каждого формата представлено его название и пример использования. Выбрать подходящий формат можно, щелкнув, на кнопке разворачивающегося списка.

Использование масок ввода и проверка значений

Если форматирование помогает варьировать представление данных в полях таблицы, то такие свойства, как Маска ввода и Условие на значение предусмотрены, чтобы не допустить попадание в таблицу некорректных данных.

Маска позволяет при вводе информации показывать в поле готовый шаблон, в который заносятся данные. Если, например, тип поля задан как текстовый, но в него должны входить только цифры (например, номер студенческого билета), можно оговорить это условие с помощью маски.

Естественен вопрос, чему отдать предпочтение: форматированию или маске ввода? Принципиальное отличие этих опций в следующем: свойство `Формат` поля применяется уже после того, как данные введены и сохранены в поле, а свойство `Маска` ввода действует еще до размещения информации в таблице, регламентируя ввод данных в конкретном поле.

Свойство `Формат` поля не влияет на значения, хранимые в таблице. Свойство `Маска` ввода используется в дополнение к формату или вместо него. Если для поля одновременно задаются формат отображения и маска ввода, то `Access` использует маску ввода при вводе или редактировании данных и свойство `Формат` поля при отображении данных после сохранения записи. В таблице 10 представлены некоторые часто используемые маски ввода и соответствующие им значения.


Таблица 10. Символы, используемые для создания масок ввода

Символ	Описание
1	2
0	Цифра (от 0 до 9, ввод обязателен; символы "плюс" [+] и "минус" [-] не допускаются).
9	Цифра или пробел (ввод не обязателен; символы "плюс" и "минус" не допускаются).
#	Цифра или пробел (ввод не обязателен; пустые символы преобразуются в пробелы, допускаются символы "+" и "-").
L	Буква (A-Z или A-Я, ввод обязателен).
?	Буква (A-Z или A-Я, ввод не обязателен).
A	Буква или цифра (ввод обязателен).
&	Любой символ или пробел (ввод обязателен).
C	Любой символ или пробел (ввод не обязателен).

. , : ; - /	Десятичный разделитель и разделители тысяч, значений даты и значений времени.
</>	Указывает перевод всех следующих символов на нижний/верхний регистр.
\	Указывает ввод любого следующего символа как постоянного символа. Используется для ввода специальных символов как постоянных символов (например, \A выводится как символ "А").
Пароль	Значение "Пароль", заданное для свойства "Маска ввода" создает поле ввода пароля. Любой символ, введенный в поле, сохраняется как символ, но отображается при вводе звездочкой (*).

Порядок действий при создании маски несложен. Первый шаг – продумать, какой должна быть маска и какие функции она будет выполнять. Второй шаг – ввести подходящий вариант маски в поле «Маска ввода». Например, для ввода номера группы может быть создана следующая маска: L"-9. Для ввода пола студента и формы оплаты за обучение допустим только один буквенный символ, поэтому используем простую маску L.

ЗАДАНИЕ 3. Задание масок и форматов ввода для таблиц БД Факультет

1. Откройте таблицу *Учетная карта* в режиме Конструктора.
2. Перейдите к полю *Nst (№ бил)* и на вкладке *Общие>Маска ввода* задайте маску **9\-99** : все символы № билета – цифры, есть обязательный разделитель “-” (тире).
3. Для поля *Ngr (№ гр)* задайте маску **L"-9** : первый символ – буква, второй – обязательное тире и третий символ – цифра.
4. Для поля *Dtr(Дата рожд.)* задайте *Краткий формат даты*, выбрав его из раскрывающегося списка на вкладке *Общие> Формат поля* (рис. 9).
5. Сохраните изменения, внесенные в структуру таблицы, выполнив команду *Файл>Сохранить* или щелкнув на кнопке *Сохранить* , и закройте таблицу.

6. Откройте в режиме Конструктора таблицу *Группы*. и задайте маску **L\9** для поля *Ngr (№ гр)*. Закройте таблицу с сохранением.
7. Откройте в режиме Конструктора таблицу *Кафедры* и задайте маску **"КФ-9"** для поля *Deprt (Кафедра)*.
8. Для поля *Naud (№ ауд)* задайте маску **L\999**. Что она означает?
9. Закройте таблицу *Кафедры*.
10. Откройте таблицу *Преподаватели* и задайте для поля *Deprt (Кафедра)* ту же маску, что и в пункте 6. Закройте таблицу с сохранением.

Проверка вводимых значений

Самый надежный способ обезопасить таблицу от ввода ошибочных или ложных данных – обеспечить их проверку на соответствие определенным требованиям непосредственно при вводе. Хотя часть работы по проверке допустимости значений данных программа Access выполняет автоматически, для обеспечения целостности данных следует предусмотреть хотя бы простейшие операции контроля.

Характер проверки задается в поле *Условие* на значение области *Свойства поля*, и если требование условия на значение не соблюдается, выдается сообщение об ошибке. Текст такого сообщения можно определить в опции *Сообщение об ошибке*.

Условие на значение вводится в виде *выражения* (команды для вычисления значений). В выражениях используются обычные символы арифметических действий: + (сложение), – (вычитание), * (умножение), / (деление). Такие символы называются *операторами*. В любом выражении присутствует, как минимум, один оператор. Значения, над которыми производится действие, называются *операндами*.

Кроме математических, для сравнения числовых величин используются *операторы сравнения*: > (больше чем), < (меньше чем), а также = (равно), <> (не равно), <= (меньше или равно), >= (больше или равно). Для операций над логическими величинами предусмотрены *логические операторы*, например, And, Or, Not.

Для текстовых значений существует оператор конкатенации (сцепления) &, с помощью которого можно объединить несколько строк символов в одну строку, и оператор Like, который проверяет соответствие строкового объекта заданному шаблону.

В выражениях можно использовать различные *функции*, например, математические, статистические или получения даты и времени. До и после даты, входящей в выражения, ставятся символы #.

ЗАДАНИЕ 4. Запись условий на значение в полях таблиц БД Факультет

1. Откройте в режиме конструктора таблицу *Учетная карта*. Выберите поле *rol(пол)*

2. Перейдите на вкладку *Общие>Условие* на значение и задайте условие **"м" Or "ж"**

3. В области ввода "Сообщение об ошибке" укажите текст **Введите м или ж.**

4. Для поля *Dtr(Дата рожд.)* задайте условие **>=#01.01.70#**, предполагая, что возраст студента дневного отделения не может превышать 33 лет (на 2003 год поступления).

5. Перейдите к таблице *Ведомость*.

6. Значение для поля, содержащего оценку за экзамен, например, для поля *1_e1* должно лежать в интервале от 2 до 5, поэтому задайте условие **>=2 and <=5** и соответствующее сообщение об ошибке **Введите целое число от 2 до 5.**

7. Задайте такое же условие на значение для поля *1_e2*.

8. Для поля *2_e1* и *2_e2* ограничьте набор допустимых значений только пятью элементами Null (пустое поле), 2, 3, 4, 5, т.е. запишите условие **Null Or 2 Or 3 Or 4 Or 5**, и дайте соответствующий комментарий в строку *Сообщение об ошибке*.

9. Для поля *3_e1* и *3_e2* запишите то же условие, используя оператор **In(Null;2;3;4;5)**.

10. Выберите любой способ для записи условий на поле *Diplom (Диплом)*, а также на поля, содержащих данные о зачетах *1_p*, *2_p* и *3_p*: (**з**–зачтено, **н**–не зачтено).

11. Проверьте все таблицы, содержащие фамилии – *Учетная карта*, *Преподаватели*, *Группы*, *Ведомость* – на наличие условия перевода всех символов на верхний регистр (рис. 7).

12. Обратите внимание, что можно работать с несколькими открытыми таблицами, переходя от одной таблицы к другой с помощью вкладки горизонтального меню **Окно**.

13. Можно расположить все таблицы, так, чтобы они были видны одновременно. Для этого на вкладке **Окно** выберите из разворачивающегося меню соответствующую форму расположения таблиц: **Сверху вниз**, **Каскадом** или **Слева направо** (рис. 10). В таком представлении таблиц можно перейти от одной таблицы к другой, щелкнув левой кнопкой в любом месте окна нужной таблицы.

14. Закройте все таблицы с сохранением структуры.

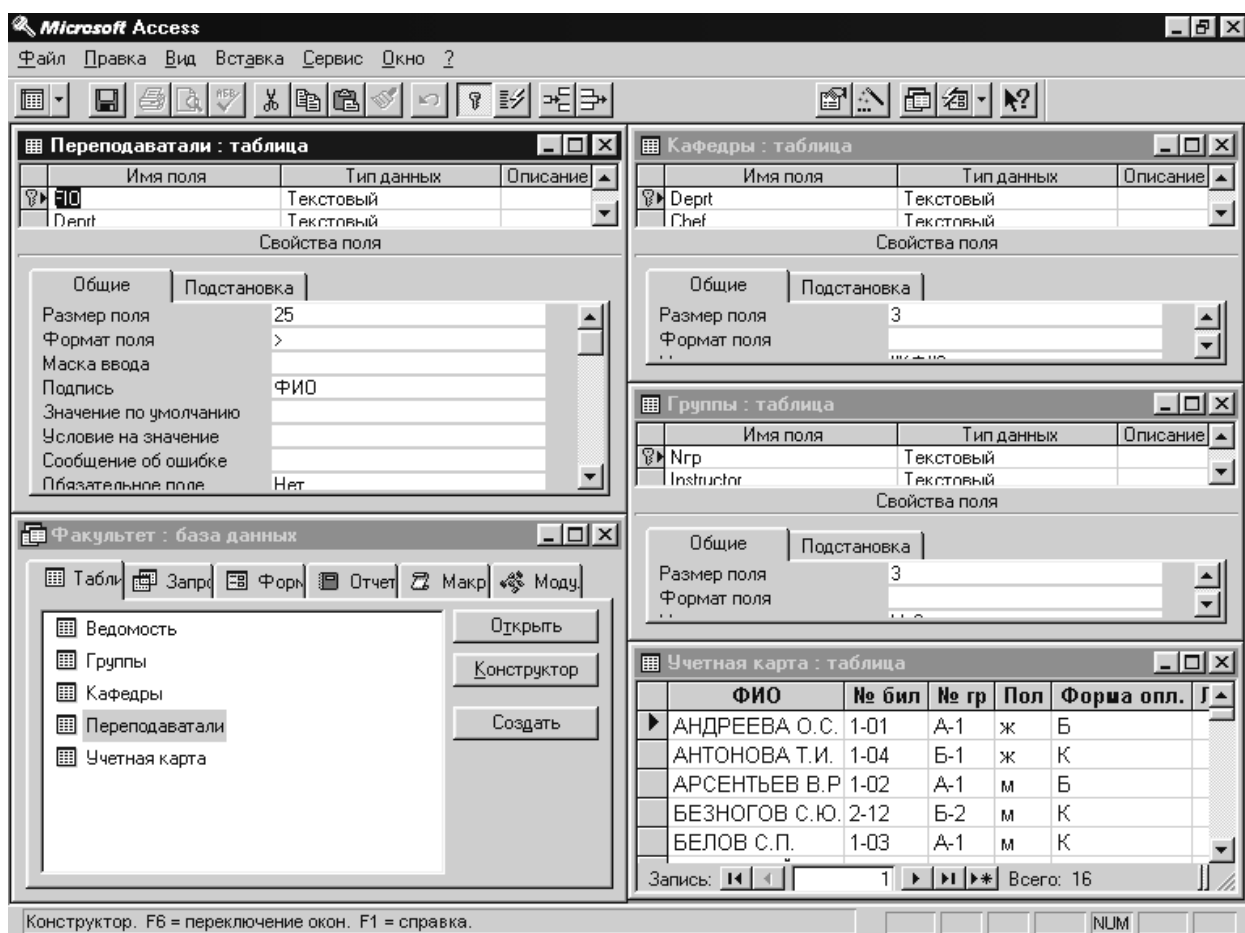


Рис.10. Расположение одновременно открытых таблиц Слева направо.

Создание простого списка подстановки

Достаточно часто бывают случаи, когда набор вводимых значений ограничен определенным списком, например, для таблицы *Преподаватели*,

требуется ввести значения в поле *Position (Должность)* пять допустимых значений: **аспирант, ассистент, доцент, ст.преп.** (старший преподаватель), **профессор**. Проверку условия на значение для этого поля можно выполнить, создав простой *список подстановки*.

ЗАДАНИЕ 5. Создание простого списка подстановок для таблицы *Преподаватели*

1. Откройте таблицу *Преподаватели* в режиме Конструктора. Выделите поле *Position*, а затем щелкните на вкладке Подстановка в области Свойства поля.

2. Щелкните на поле Тип элемента управления, а затем выберите из раскрывающегося списка пункт Поле со списком или Список. Теперь на вкладке Подстановка появился новый набор опций, которые необходимо определить.

3. Щелкните на поле Тип источника строк и выберите из раскрывающегося списка пункт Список значений. Оставьте неизменным принятое для свойства поля Присоединенный столбец и Число столбцов значение 1.

4. Для свойств Ширина столбцов введите значение 1 см, а для свойства Ширина списка – Авто. Так как число опций списка равно 5, укажите значение 5 для свойства Число строк списка (по умолчанию - 8).

5. Для свойства Ограничиться списком укажите значение **Да**, тем самым ограничив число возможных значений для этого поля только теми, которые будут перечислены в списке подстановки.

6. Теперь введите элементы будущего списка в поле Источник строк в следующем виде: **аспирант; ассистент; доцент; ст.преп.; профессор**. Элементы ввода разделены символами точки с запятой: рис. 11.

8. *Должность* можно открыть список подстановки (рис. 12 а).

9. Откройте в режиме Конструктора таблицу *Учетная карта* и создайте список подстановки для поля *Форма Опл.* с двумя столбцами согласно рис. 12 б). Сохраните изменения.

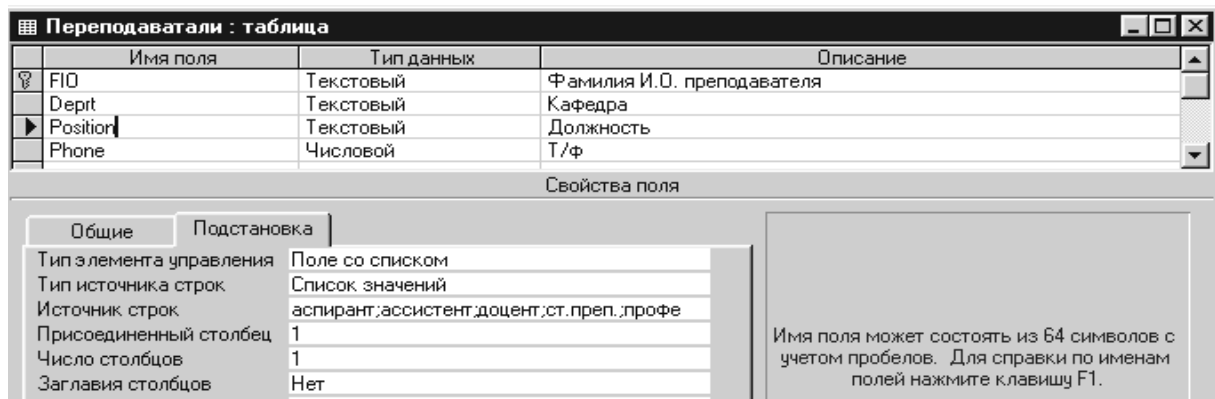


Рис. 11. Для поля *Должность* создается список подстановки.
 Все значения для опций списка вводятся во вкладке
 Подстановка области Свойства поля

10.Теперь, если перейти в режим таблицы и переместиться в поле

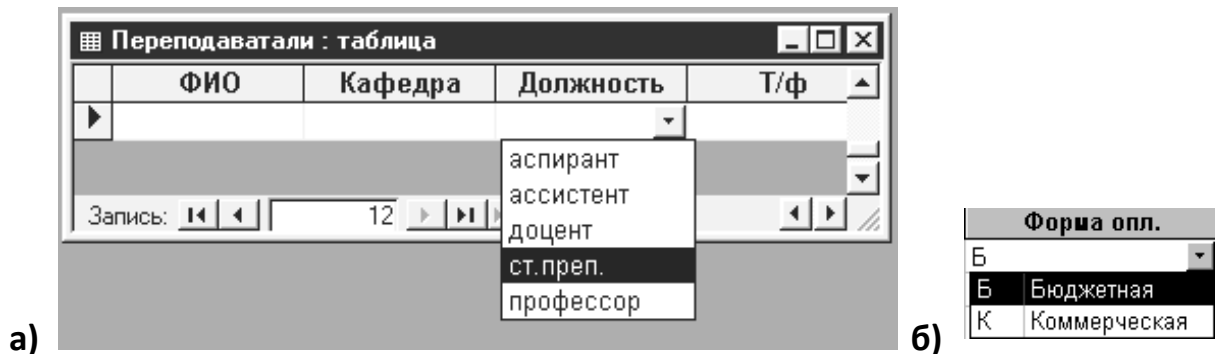


Рис. 12. а) Список подстановки для поля *Должность* включает перечень допустимых значений. б) Список подстановки для поля *Раум(Форма опл.)* таблицы *Учетная карта*

Другие опции области Свойства поля

Прежде чем приступать к окончательному вводу данных в таблицы БД *Факультет*, рассмотрим другие свойства поля, которые активно используются при настройке структуры таблиц базы данных.

- Свойство *Значение по умолчанию*. Эта возможность позволяет упростить и ускорить ввод одинаковых значений в поле, а в случае необходимости вносить исправления уже в готовое значение, которое автоматически будет появляться в этом поле перед вводом данных. Таким значением может быть выражение, число или текст, которое соответствует типу данных поля. Для числовых и денежных данных по умолчанию устанавливается значение, равное 0.

- Свойство *Обязательное значение*. Определяет, требуется ли обязательно вводить данные в это поле. Если для свойства установлено значение «Да», при вводе или редактировании записей в таблице программа Access не «отпустит» вас, пока в этом поле не появятся данные. Пустые (Null) значения в этом поле не допускаются.

Ввод данных в таблицы

Для того, чтобы БД *Факультет* могла функционировать нормально, необходимо в каждую таблицу ввести необходимый минимум данных. На рис. 13 – 16 представлены данные для четырех таблиц БД (данные таблицы *Учетная карта* представлены на рис. 6).

ЗАДАНИЕ 6. Ввод данных в таблицы

1. Откройте таблицу *Учетная карта* в режиме *Таблицы* и продолжите набор данных согласно рис. 13. Обратите внимание, как работают маски в полях *№ бил* и *№ гр*. Для поля *Форма опл.* воспользуйтесь созданным в предыдущем **ЗАДАНИИ 5** списком подстановки.

2. Откройте таблицу *Ведомость* в режиме *таблицы*.

В этом задании при заполнении поля *ФИО* таблицы *Ведомость* (а также и других таблиц) используйте прием *копирования данных через буфер обмена*. Этот способ не очень удобен, но на данном уровне знакомства с Access пока что является единственно доступным для ускорения ввода данных.

Познакомившись с приемами связывания таблиц, создания запросов и форм, можно будет использовать более эффективные способы заполнения таблиц.

Итак, чтобы ускорить ввод данных и соблюсти однозначное соответствие записей в ключевом поле, перенесем данные в поле *ФИО* из таблицы *Учетная карта*, используя буфер обмена. Для этого выполните следующие действия.

3. Откройте таблицу *Учетная карта* и в опции *Окно* выберите расположение *Слева направо*.

4. Выделите столбец *ФИО*, щелкнув по названию поля. Выполните команду *Правка>Копировать*.

5. Перейдите к полю *ФИО* таблицы *Ведомость*. Выполните команду *Правка>Добавить* из буфера. На эти действия Access выдаст предупреждение (рис. 17).

ФИО	1_р	1_е1	1_е2	2_р	2_е1	2_е2	3_р	3_е1	3_е2	Диплом
АНДРЕЕВА О.С.	з	4	3з		3	4				
АНТОНОВА Т.И.	з	5	4з		4	5				
АРСЕНТЬЕВ В.Р.	з	4	4з		4	4				
БЕЗНОГОВ С.Ю.	з	5	5з		4	4				
БЕЛОВ С.П.	з	5	3з		5	4				
БЕРЕЖНОЙ П.Р.	з	5	5з		5	5				
ВЕРХОВ А.В.	н	3	2з		3	3				
ГАЛКИН И.Д.	з	4	4з		4	5				
ДЕНИСОВ И.О.	з	4	4з		3	5				
ДРАГУНОВ А.А.	з	4	4з		5	5				
ЕЛИСЕЕВА И.В.	з	5	4з		3	3				
ЛЕОНТЬЕВ М.В.	з	3	4з		3	3				
РОМАНОВА С.Л.	з	3	4з		3	4				
СЕМЕНОВ А.Л.	з	4	5з		5	4				
ТОРОВА Р.К.	з	3	5з		4	3				
ЯКОВЛЕВ С.И.	з	4	3з		5	4				
*					-1					

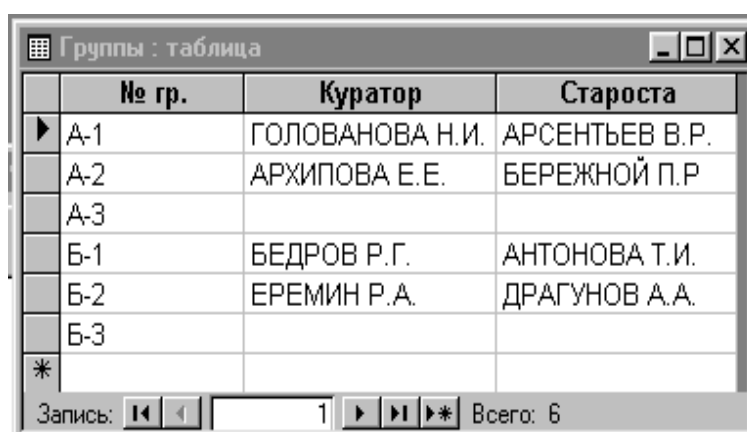
Рис.13. Таблица *Ведомость* открыта в режиме Таблицы для ввода записей

ФИО	Кафедра	Должность	Т/ф
АНТОНОВ В.А.	КФ-1	профессор	361456
АРХИПОВА Е.Е.	КФ-2	доцент	710880
БЕДРОВ Р.Г.	КФ-1	доцент	823261356
ВЕРЕСОВ Е.А.	КФ-3	профессор	725684
ГОЛОВАНОВА Н.И.	КФ-1	ст. преп.	642286
ДАНИЛОВ А.И.	КФ-2	профессор	564278
ЕРЕМИН Р.А.	КФ-2	ассистент	833225481
ЗАЙЦЕВ А.С.	КФ-3	профессор	710153

Рис.14. Таблица *Преподаватели* открыта в режиме Таблицы для ввода записей

6. Подтвердите вставку записей. В результате таблица *Ведомость* будет иметь столько же записей, что и *Учетная карта*.

7. Перейдите в режим конструктора и проставьте в полях, где должны быть введены результаты зачетов (1_р, 2_р, 3_р) Значение по умолчанию "з".



	№ гр.	Куратор	Староста
▶	А-1	ГОЛОВАНОВА Н.И.	АРСЕНТЬЕВ В.Р.
	А-2	АРХИПОВА Е.Е.	БЕРЕЖНОЙ П.Р.
	А-3		
	Б-1	БЕДРОВ Р.Г.	АНТОНОВА Т.И.
	Б-2	ЕРЕМИН Р.А.	ДРАГУНОВ А.А.
	Б-3		
*			

Запись: [←] [←] [1] [→] [→] [→*] Всего: 6

Рис.15. Таблица *Группы* открыта в режиме Таблицы для ввода записей

Кафедры : таблица				
	Кафедра	Зав.каф.	№ ауд	Т/ф
▶	КФ1	АНТОНОВ В.А.	А-121	435656
	КФ2	ДАНИЛОВ А.И.	Б-403	433247
	КФ3	ЗАЙЦЕВ А.С.	А-225	436428
*				0

Запись: [←] [←] [1] [→] [→] [→*] Всего: 3

Рис.16. Таблица *Кафедры* открыта в режиме Таблицы для ввода записей

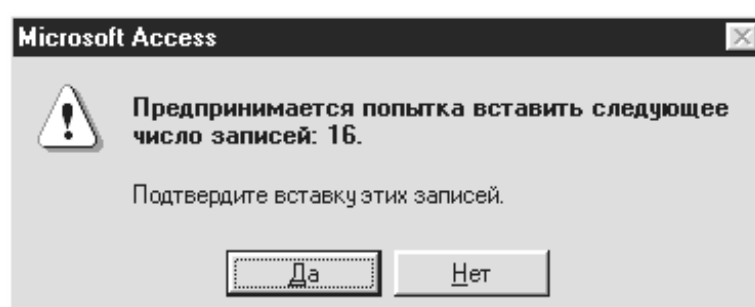


Рис. 17. Предупреждение Access о попытке вставить записи

8. Перейдите в режим таблицы и, сравнивая таблицу с рис. 13, исправьте те записи, где стоит "н".

9. Введите остальные оценки. Закройте таблицы *Учетная карта* и *Ведомость*.

10. Откройте таблицу *Преподаватели* в режиме Таблицы и введите в нее данные согласно рис. 14. Обратите внимание, как работает маска в поле *Кафедра*. Воспользуйтесь списком подстановок для ввода должности преподавателя.

11. Откройте таблицу *Группы* в режиме Таблицы. Заполните ее согласно рис. 15. Можете воспользоваться буфером обмена для ввода фамилий кураторов и старост, копируя в него соответствующие фамилии из таблиц *Учетная карта*, *Преподаватели*.

12. Откройте таблицу *Кафедры* в режиме Таблицы. Заполните ее согласно рис. 16. Обратите внимание, как работает маска в поле *Кафедра*. Фамилии в поле *Зав.каф.* можно копировать из таблицы *Преподаватели*.

13. Закройте все таблицы. Обратите внимание, потребует ли программа от вас процедуры сохранения? Почему?

Итак, на данном этапе создания базы данных вы обеспечили контроль и форматирование данных на этапе ввода, создали списки подстановок, научились работать сразу с несколькими таблицами, видоизменяя их расположение в окне. Кроме того, вы заполнили таблицы базы данных значениями, относящимися к студентам II и III курсов. Ввод значений для студентов I курса будем осуществлять позднее, используя более эффективные методы работы с программой Access.

Организация связей между таблицами

Когда между таблицами устанавливается связь, это означает, что величины из одной таблицы ставятся в соответствие величинам из другой таблицы.


Нами уже были рассмотрены определения таких важных понятий, как *первичный* и *внешний ключи*, вопрос выбора первичного ключа и рассмотрен основной принцип, позволяющий реализовать связь между реляционными таблицами: такая связь становится возможной благодаря наличию в этих таблицах полей с совпадающими значениями данных.


Существуют условия, которым должны соответствовать типы данных связываемых полей.

- Общие или связываемые поля должны быть одинакового типа.
- Если оба связываемых поля имеют числовой тип, они должны иметь и одинаковые значения свойства `Размер поля`.
- Поле с типом `Счетчик` можно связать с числовым полем, у которого свойство `Размер поля` имеет значение `Длинное целое`.

Поскольку при проектировании БД *Факультет* сразу предполагалось, что все таблицы этой БД будут связаны между собой, рассмотрим в качестве примера создание таких связей. Для этого необходимо открыть окно *Схема данных* и добавить в него те таблицы, которые будут связаны. Окно *Схема данных* – это удобное и наглядное графическое представление связей для открытой базы данных, где можно добавлять таблицы, определять тип связи и устанавливать параметры целостности данных между таблицами. Итак, следуя рекомендациям **ЗАДАНИЯ 7**, установите связи между таблицами БД *Факультет*.

ЗАДАНИЕ 7. Установка связей между таблицами

1. Открыть базу данных и выполнить команду горизонтального меню Сервис>Схема данных или щелкнуть на кнопке  Схема данных панели инструментов. Откроется диалоговое окно Схема данных. Если БД уже открыта, то создавать или изменять связи между открытыми таблицами нельзя, поэтому перед началом работы их следует закрыть.

2. В том случае, если в окно Схема данных еще не добавлены никакие таблицы, сразу откроется окно Добавление таблицы. Если оно не появится, выполните команду горизонтального меню Связи>Добавить таблицу или щелкните на кнопке  Добавить таблицу на панели инструментов.

3. Дважды щелкните на имени таблицы (например, *Учетная карта*), которая должна быть включена в окно схемы данных.

4. Выполните шаг 3 требуемое число раз, чтобы включить в рабочую область окна Схемы данных все необходимые таблицы. Затем щелкните на кнопке Закреть. Для нашего примера таких таблиц будет пять, и в итоге все они появятся в окне Схема данных в виде отдельных окон, где перечислены поля, содержащиеся в таблицах (рис. 18).

5. Создайте теперь связь между таблицами *Учетная карта* и *Ведомость*. Как отмечалось ранее, это связь типа «один к одному». Каждой записи первой таблицы соответствует одна (и не более) связанная запись второй таблицы (и, соответственно, наоборот).

В обеих таблицах роль первичного ключа выполняет поле *ФИО*. На этом поле таблицы *Учетная карта* нажмите левую клавишу мыши и, удерживая ее, перетащите на поле *ФИО* таблицы *Ведомость*. Затем отпустите левую кнопку мыши. (При таком порядке установления связи первую таблицу *Учетная карта* можно рассматривать как главную, а *Ведомость* – как подчиненную).



Рис.18. В окне *Схема данных* отображаются пять таблиц базы данных *Факультет*

6. После того, как поле перенесено, появится диалоговое окно *Изменение связей* (рис. 19).

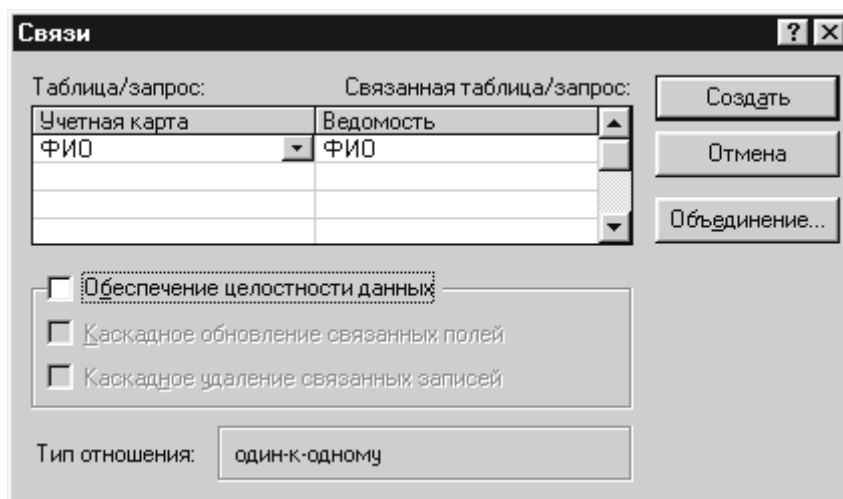


Рис. 19. Диалоговое окно *Изменение связей* позволяет определить параметры создаваемой связи

7. Щелкните в этом окне на кнопке *Создать*, в результате между двумя полями появится тонкая *линия объединения* (рис. 20).

8. Теперь создайте связь между таблицами *Учетная карта* и *Группы*. Здесь, как уже отмечалось, связь типа «один ко многим». Одна из таблиц – *Группы* – называется родительской, или главной (общее поле *Ngr* является в ней первичным ключом), а вторая – дочерней, или подчиненной (поле *Ngr* в ней – это внешний ключ). Теперь, как и в п.5, первичный ключ надо выделить и совместить со вторичным, установив тем самым связь между указанными таблицами.

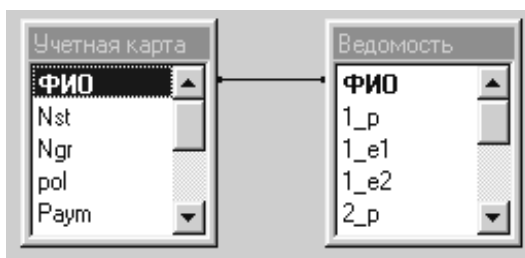


Рис. 20. Таблицы *Учетная карта* и *Ведомость* теперь связаны между собой

9. Теперь необходимо повторить шаги 5 – 7 для каждой пары таблиц, которые необходимо связать между собой. Сделайте это самостоятельно, получив в результате этих операций в окне *Схема данных* структуру связей, изображенную на рис. 21.

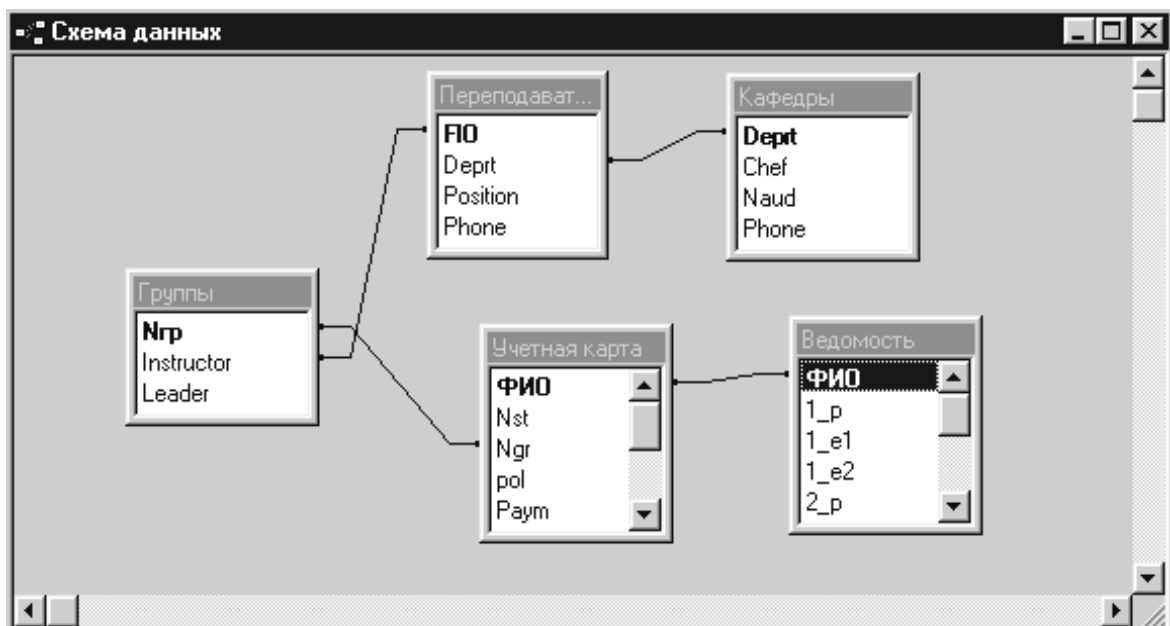


Рис. 21. В окне *Схема данных* отображены все связи между таблицами базы данных *Факультет*

10. Сохраните созданные связи, используя стандартные приемы Windows для сохранения. Но, даже, если Вы не сделаете этого, программа Access предложит вам сохранить созданные связи и при закрытии окна *Схема данных*.

Обеспечение целостности данных в базе

При определении связи между полями двух таблиц в диалоговом окне Изменение связей имеется опция Обеспечение целостности данных (рис. 19). *Целостность данных* – это система правил, регулирующих взаимодействие между связанными таблицами и обеспечивающих корректность хранимых в таких таблицах данных. Целостность данных подразумевает, что данные, введенные в общее поле двух связанных таблиц, должны совпадать.

Если используются условия целостности данных, то Access будет поддерживать следующие правила.

- В поле внешнего ключа дочерней таблицы нельзя ввести значение, не содержащееся в поле первичного ключа родительской таблицы. Таким образом, если попытаться указать в поле *Ngr* номер группы, отсутствующий в списке *Группы* (поле *Ngr*), при сохранении записи программа Access выдаст сообщение об ошибке.

- Программа Access не позволяет удалить запись родительской таблицы, если в дочерней таблице имеются связанные с ней записи. Например, нельзя удалить из таблицы *Преподаватели* сведения о преподавателе, если он является куратором и о нем имеется запись в таблице *Группы*.

- Нельзя изменить значение первичного ключа в родительской таблице, если для данной записи существуют связанные записи в дочерней таблице. Например, если попытаться изменить название кафедры в поле *Deprt* таблицы *Кафедры*, то это сделать будет невозможно, т.к. в таблице *Преподаватели* существуют записи, в которых используется старое название (рис. 22).

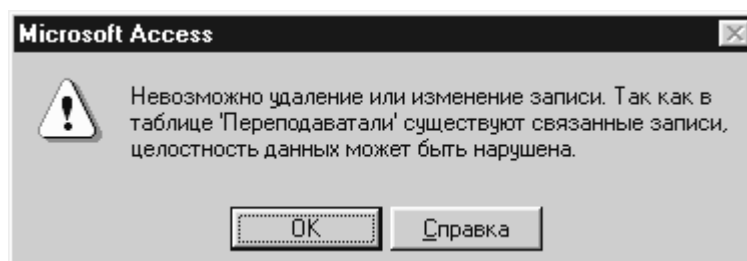


Рис. 22. Предупреждение Access о невозможности изменения данных, которое может нарушить их целостность

При выборе опции Обеспечения целостности в окне Изменение связей (рис. 19) становятся доступными еще две опции, которые позволяют активизировать автоматическое выполнение каскадного удаления и каскадного обновления данных.

- Если поставить флажок на опции Каскадное обновление связанных полей, то при любом изменении данных первичного ключа в родительской таблице (например, названия кафедры в поле *Dept* таблицы *Кафедры*) автоматически будут обновляться соответствующие значения в поле связанной таблицы (поле *Dept* таблицы *Преподаватели*). Благодаря внесению необходимых изменений в связанную таблицу целостность данных не будет нарушена. Если эта опция не включена, изменить значение ключевого поля первичной таблицы не удастся, как это уже было показано в предыдущем пункте.

- Когда установлен флажок в опции Каскадное удаление связанных полей, то при удалении записи в родительской таблице автоматически будут удаляться соответствующие записи в дочерней связанной таблице. Например, если такого рода связь установлена между таблицами *Учетная карта* и *Ведомость*, то при удалении записи из одной таблицы *Учетная карта* автоматически будет удалена такая же запись из таблицы *Ведомость*.

ЗАДАНИЕ 8. Редактирование и удаление связей между таблицами

В предыдущем **ЗАДАНИИ 7** между всеми таблицами БД *Факультет* были установлены связи. Любая связь может быть отредактирована или удалена. Для это необходимо выполнить следующие действия:

1. Если таблицы, между которыми необходимо изменить связь, открыты, заройте их.

2. Открыть окно Схема данных.

3. В окне Схема данных подвести курсор к линии связи, которую надо откорректировать, и двойным щелчком открыть диалоговое окно Изменение связей (рис. 19).

4. Внесите необходимые изменения: установите целостность данных и каскадное обновление полей, проставив в соответствующих окнах флажки. После закрытия окна Изменение связей вы снова вернетесь в окно Схема данных.

5. Обратите внимание, как изменился вид линий связи: теперь по схеме данных можно определить тип связи (рис. 23).

6. После установления целостности связей всех таблиц откройте в режиме таблицы таблицу *Учетная карта* и *Ведомость*. Введите в первую

таблицу в поле ФИО фамилии **Иванов И.И.** и **Петров П.П.**, другие поля заполнять необязательно (запись считается введенной, если курсор уже переведен в поле временной (новой) записи). Затем через буфер обмена перенесите эти фамилии в таблицу *Ведомость*.

7. Установите расположение окон Слева направо, используя пункт горизонтального меню *Окно*. Перейдите в таблицу *Учетная карта*. Удалите запись **Иванов И.И.** (используйте пункт *Правка>Удалить* запись горизонтального меню). Внимательно прочтите предупреждение, которое выдает Access на ваши действия (рис. 24 а). Подтвердите удаление. Обратите внимание, что произошло с этой же записью в таблице *Ведомость*.

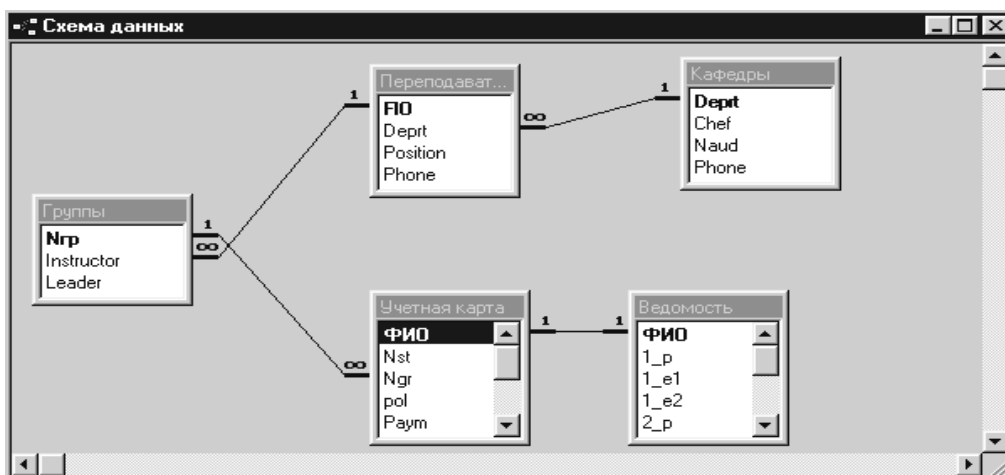
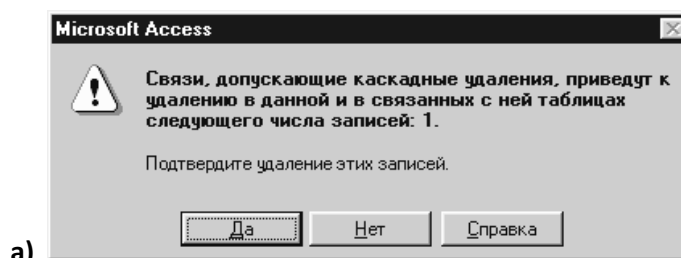


Рис. 23. Схема связей между таблицами, в которых обеспечивается целостность данных

8. Установите курсор в таблицу *Ведомость*. Удалите запись с фамилией **Петров П.П.** Обратите внимание на предупреждение, сделанное Access в этом случае (рис. 24 б). Самостоятельно удалите запись с фамилией **Петров П.П.** из таблицы *Учетная карта*. Закройте обе таблицы.



а)

9. Откройте таблицы *Группы* и *Преподаватели*. и расположите их Слева направо.

10. Попробуйте удалить запись с фамилией **Голованова Н.И.** из таблицы *Преподаватели*. Access не может позволить вам этого сделать, иначе нарушается целостность данных (скопируйте запись с фамилией **Голованова Н.И.** в буфер обмена).

11. Удалите фамилию **Голованова Н.И.** из поля *Кураторы* таблицы *Группы*. А затем еще раз повторите шаг 10. На этот раз удаление записи должно пройти успешно.

12. Вставьте из буфера удаленную запись в таблицу *Преподаватели* и впишите эту же фамилию в поле *Кураторы* таблицы *Группы*. Закройте все таблицы.

Если понадобится удалить межтабличную связь, например, для того, чтобы изменить тип данных поля, участвующего в связи, необходимо выполнить следующие действия.

1. Щелкнуть на требуемой линии объединения в окне *Схема данных* (линия при этом станет более жирной) и нажать клавишу <Delete>. Есть и другой способ: щелкнуть правой кнопкой мыши на нужной линии связи и выбрать из появившегося контекстного меню команду *Удалить*.

2. На экране появится диалоговое окно Access, в котором необходимо будет подтвердить удаление связи.

Итак, таблицы баз данных можно связывать между собой, используя общее поле. На данном этапе конструирования базы вы изучили, как создавать, редактировать и удалять связи между таблицами, при необходимости указывая опции обеспечения целостности данных, такие, как каскадное обновление связанных полей и каскадное удаление связанных записей.

Благодаря связям между таблицами можно создавать запросы, формы и отчеты, отображающие информацию из нескольких таблиц. Изучению этих возможностей Access и будут посвящены следующие задания.

Контрольные вопросы

1. Что такое база данных с точки зрения ее хранения на диске?
2. Как создать файл новой базы данных?
3. Можно ли изменить структуру таблиц после их сохранения? В чем заключаются эти изменения?
4. Что означает сохранение данных в режиме реального времени?
5. Что такое временная запись таблицы? Можно ли разместить новую запись между уже существующими записями или поместить ее в начале таблицы?
6. Какова реакция программы Access на ввод буквенных символов в поле числового типа?
7. Как осуществляется копирование записей через буфер обмена?
8. С какой целью делается форматирование данных в поле?
9. Какой специальный формат необходимо применить, чтобы все символы текстового поля изображались заглавными буквами?
10. Для каких типов полей разработаны встроенные форматы?
11. Для чего предусмотрено свойство Маска ввода? В чем принципиальное отличие опций форматирования и маски ввода?
12. С помощью какого свойства поля можно организовать проверку вводимых значений в числовое поле?
13. Из каких элементов складывается выражение при проверке условия на значение?

14. Для чего используется список подстановки?

15. Какие возможности предоставляют свойства поля Значение по умолчанию, Обязательное значение?

16. Когда Access потребует сохранения таблицы: при изменении свойств ее полей или при вводе данных? Что при этом сохраняется: значения в полях таблицы или ее структура?

17. Каким условиям должны соответствовать типы данных связываемых полей? Поля каких ключей используются в роли общего поля при реализации связи между реляционными таблицами?

18. Каковы условия поддержки целостности данных при связывании реляционных таблиц?

19. Как изменяются данные в ключевых полях, если при установлении целостности данных установлено Каскадное обновление связанных полей?

Оформление отчета

Отчет о лабораторной работе должен содержать:

1. Описание в тетради основных этапов проделанной работы, их логическую последовательность.
2. Ответы на контрольные вопросы.

