

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
“БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТРАНСПОРТА”

Кафедра "Информационные технологии"

Т. Л. ШИНКЕВИЧ, Т. Н. ЛИТВИНОВИЧ, Ю. П. ЛЫЧ

**СИСТЕМА УПРАВЛЕНИЯ
БАЗАМИ ДАННЫХ MS ACCESS**

Лабораторный практикум по курсу «ТОХОД»

Часть 1

Гомель 2004

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
“БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТРАНСПОРТА”

Кафедра "Информационные технологии"

Т. Л. ШИНКЕВИЧ, Т. Н. ЛИТВИНОВИЧ, Ю. П. ЛЫЧ

СИСТЕМА УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ MS ACCESS

Лабораторный практикум по курсу «ТОХОД»

Часть 1

*Одобен методической комиссией
гуманитарно-экономического факультета*

Гомель 2004

УДК 681.3(076.5)
Ш 626

Рецензент – **Ю. А. Пшеничнов**, канд. техн. наук, доцент кафедры «Информационные технологии».

Шинкевич Т. Л., Литвинович Т. Н., Лыч Ю. П.

Ш 626 Система управления базами данных MS ACCESS Ч. 1: Лабораторный практикум по курсу «ТОХОД» *Шинкевич Т.Л., Литвинович Т.Н., Ю.П. Лыч* – Гомель: БелГУТ, 2004. – 36 с.

Практикум является продолжением серии методических указаний по информатике, и охватывает следующие разделы: проектирование баз данных, создание таблиц и форм баз данных. Предназначен для студентов дневной формы обучения ГЭФ.

УДК 681.3(073.5)

© Т. Л. Шинкевич, Т. Н. Литвинович, Ю. П. Лыч, 2004.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
<i>Лабораторная работа № 1 ПРОЕКТИРОВАНИЕ И СОЗДАНИЕ БАЗЫ ДАННЫХ</i>	<i>4</i>
СОЗДАНИЕ ТАБЛИЦ БАЗЫ ДАННЫХ	4
1 ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ	4
1.1 Проектирование базы данных «Торговля»	4
1.2 Запуск MS ACCESS	6
2 ТАБЛИЦЫ	6
2.1 Создание таблицы	7
2.1.1 Создание таблицы в режиме Конструктора	9
2.2 Определение имен полей	9
2.3 Типы данных	10
2.4 Заполнение полей таблицы в окне конструктора	10
2.5 Свойства полей	11
2.5.1 Задание маски ввода	12
2.5.2 Задание свойства условия на значение	13
2.6 Выбор ключевого поля	13
2.6.1 Свойства ключевого поля	14
2.7 Сохранение таблицы	14
2.8 Присвоение имени объектам	14
2.9 Межтабличные связи	14
2.9.1 Создание схемы данных	15
2.9.2 Задание условий целостности	16
2.10 Ввод данных в таблицу	17
2.11 Перемещение по таблице	18
2.12 Редактирование данных в таблице	19
2.13 Изменение структуры таблицы	20
2.14 Сжатие данных	20
2.15 Внешний вид таблицы	20
Порядок выполнения работы	20
<i>Лабораторная работа № 2 СОЗДАНИЕ ФОРМ</i>	<i>23</i>
1 ФОРМЫ	23
1.1 Создание формы	23
1.1.1 Создание автоформы	24
1.2 Ввод и редактирование данных в форме	25
1.3 Мастер форм	25
1.3.1 Создание подчиненной формы	27
1.4 Конструктор форм	29
1.4.1 Изменение структуры формы	30
1.5 Элементы управления	30
1.5.1 Перемещение элементов управления	33
1.5.2 Изменение размеров элементов управления	33
1.5.3 Создание заголовка формы	33
1.5.4 Задание стиля оформления элементов управления	34
1.5.5 Изменение расположения элементов управления	34
1.5.6 Добавления элемента управления в форму	34
1.5.7 Создание элементов управления	34
Порядок выполнения работы	35
СПИСОК ЛИТЕРАТУРЫ	36

ВВЕДЕНИЕ

База данных – это организованная структура, предназначенная для хранения информации.

Кроме данных, база содержит методы и средства, позволяющие пользователю оперировать только с теми данными, которые входят в его компетенцию. В результате взаимодействия данных, содержащихся в базе, с методами, доступными конкретным пользователям, образуется информация, которую они потребляют и на основании которой в пределах собственной компетенции производят ввод и редактирование данных.

С понятием базы данных тесно связано понятие системы управления базой данных. Это комплекс программных средств, предназначенных для создания структуры новой базы, наполнения ее содержанием, редактирования содержимого и визуализации информации (отбор отображаемых данных в соответствии с заданным критерием, их упорядочение, оформление и последующая выдача на устройство вывода).

Система управления базами данных Microsoft Access является реляционной СУБД для операционной системы Microsoft Windows.

В MS Access предусмотрены все необходимые средства для определения и обработки данных, а также для управления ими при работе с большими объемами информации. Основными объектами Access являются таблицы, запросы, формы, отчеты, макросы и модули.

Таблицы – это основные объекты базы данных. В таблицах хранятся все данные, имеющиеся в базе, и структура базы (поля, их типы и свойства).

Формы – это средства для ввода данных. С помощью форм данные можно не только вводить, но и отображать.

Запросы – это объекты, которые служат для извлечения данных из таблиц и предоставления их пользователю в удобном виде.

Отчеты по своим свойствам и структуре во многом похожи на формы, но предназначены только для вывода данных, причем для вывода не на экран, а на печатающее устройство (принтер).

Макросы и модули – эти категории объектов предназначены как для автоматизации повторяющихся операций при работе с системой управления базами данных, так и для создания новых функций путем программирования.

ПРОЕКТИРОВАНИЕ И СОЗДАНИЕ БАЗЫ ДАННЫХ СОЗДАНИЕ ТАБЛИЦ БАЗЫ ДАННЫХ

Цель работы: запуск и работа в окне ACCESS. Открытие базы данных. Создание базы данных: создание файла базы данных, создание таблицы базы данных.

1 ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ

Разработка любой базы данных начинается с ее проектирования.

Продуманная структура и правильно выбранный формат записей способствуют эффективной работе с базой данных и позволяют быстро получить доступ к необходимой информации.

На этапе проектирования необходимо:

- 1 Определить назначение базы данных.
- 2 Принять решение о том, какие исходные данные (таблицы) база должна содержать.
- 3 Определить поля, которые будут входить в таблицы, типы полей и их свойства.
- 4 Выбрать поля, содержащие уникальные значения.
- 5 Распределить данные по таблицам.
- 6 Назначить связи между таблицами.

1.1 Проектирование базы данных «Торговля»

Имеются следующие сведения о магазинах и поставках товаров в эти магазины: № магазина, телефон магазина, адрес магазина, фамилия, имя, отчество сотрудников магазина, должность сотрудника, наименование товаров, которые поступают в магазины, цена товаров, дата поступления товаров, количество поступающих товаров.

Необходимо с помощью СУБД ACCESS разработать структуру базы данных, отражающую реальную ситуацию о поступлении товаров в магазины. Для этого необходимо:

1 Распределить данные по таблицам и определить уникальные поля. В таблице «Магазины» будут содержаться сведения: № магазина, адрес магазина, номер телефона магазина. В этой таблице каждому магазину соответствует только один адрес и телефон. Следовательно, поле *№ магазина* можно определить как уникальное (ключевое) поле.

В таблице «Сотрудники» будут содержаться сведения о сотрудниках, работающих в магазинах: фамилия, имя, отчество, должность. Чтобы можно было связать эту таблицу с другими таблицами, добавим в эту таблицу поле *№ магазина*. При этом каждому магазину в этой таблице будет соответствовать несколько фамилий сотрудников (директора, товаровед и т.д.). Следовательно, ни одно из имеющихся в этой таблице полей не может быть уни-

кальным. Поэтому вводим фиктивное поле № по порядку и определим его как ключевое.

В таблице «Товары» будут содержаться сведения о товарах: наименование товара и цена товара. Каждому товару поставим в соответствие код. Поле код товара будет уникальным в данной таблице. Следовательно, в этой таблице ключевым полем можно выбрать поле код товара.

В таблице «Поступление» будут содержаться сведения о поступлении товаров в магазины: № магазина, дата поступления товаров в магазины, код товара, количество товаров. В этой таблице один и тот же код товара может встречаться несколько раз, поскольку один и тот же товар может поступать в разные магазины и по разным датам. В этой таблице нет такого поля, которое могло бы служить ключом, поэтому введем в таблицу фиктивное поле № по порядку и определим его как ключевое.

2 Для каждого поля, включаемого в запись, установить тип данных, определяющий вид информации, которая будет храниться в поле. Поле телефон объявим текстовым полем, а не числовым. Объясняется это тем, что в записи номеров телефонов встречаются специальные символы, такие как дефис и скобки, например (023)-295-39-97. В числовых же полях использовать другие символы, кроме цифр, точек и запятых, не допускается. Данные в поле счетчика всегда уникальны по определению, поэтому во всех таблицах ключевым полям присвоим тип данных «счетчик».

3 Установить связи между таблицами. Из таблицы «Магазины» ключевое поле № магазина свяжем с соответствующим полем таблицы «Сотрудники» (в этой таблице № магазина – не ключевое поле) – тип связи «один-ко-многим». Таблицы «Товары» и «Поступление» свяжем по полю код товара (тип связи «один-ко-многим»). Таблицы «Магазины» и «Поступление» свяжем по полю № магазина (тип связи «один-ко-многим»). Разработанная структура базы данных «Торговля» представлена на рисунке 1.1.

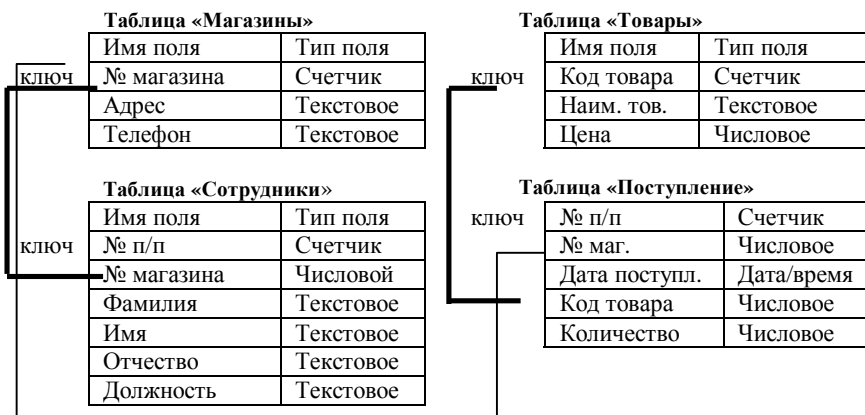


Рисунок 1.1 – Структура базы данных «Торговля»

1.2 Запуск MS ACCESS

Для запуска MS ACCESS в главном меню Windows (кнопка «Пуск») необходимо выбрать **Программы**→**MS Access**. Откроется окно (рисунок 1.2), в котором находятся две основные опции для создания новой и открытия существующей базы данных.

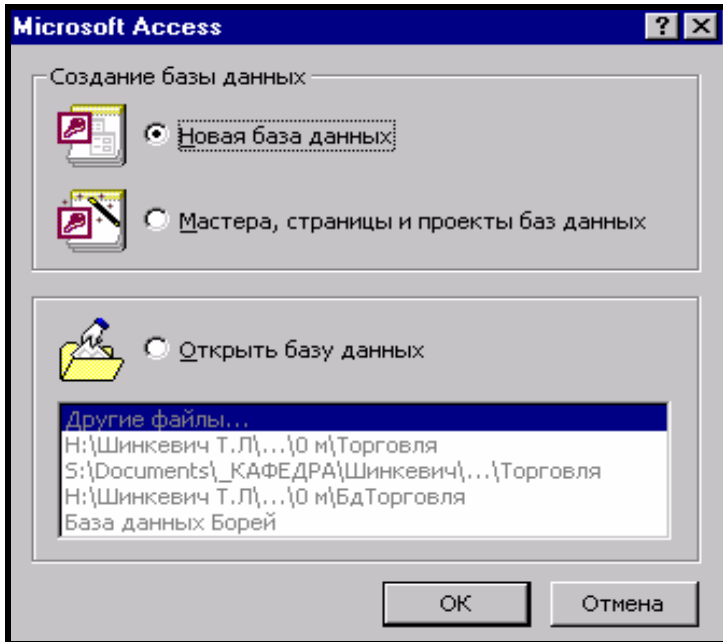


Рисунок 1.2 – Окно-приглашение MS Access

Для создания новой базы данных необходимо выбрать опцию «Новая база данных». Нажать кнопку **ОК**. Откроется окно (рисунок 1.3), в котором необходимо указать место сохранения базы данных, имя базы данных и нажать кнопку **Создать**. Сохранить файл базы данных надо до того, как приступите к созданию таблиц. Таким образом, создается пустая база данных, в которую впоследствии можно поместить таблицы, формы, запросы, отчеты.

2 ТАБЛИЦЫ

В ACCESS вся информация хранится в виде таблиц. Таблицей является совокупность данных, объединенных общей темой. Для каждой темы отводится отдельная таблица, что позволяет избежать повторения сохраняемых данных. Каждая строка таблицы представляет собой запись, а столбец – по-

ле. Запись содержит набор данных об одном объекте: например, № магазина, его адрес и телефон, а поле – однородные данные о всех объектах (например, адреса всех занесенных в таблицу магазинов). Количество полей в записи, и их тип определяются в процессе проектирования базы данных.

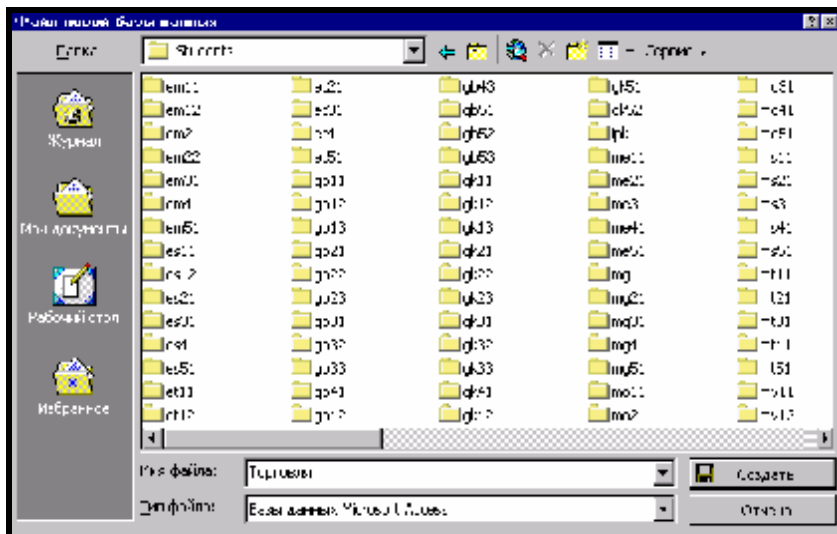


Рисунок 1.3 – Создание файла базы данных

2.1 Создание таблицы

Для создания таблицы необходимо открыть базу данных и на вкладке **Таблица** щелкнуть на кнопке **Создать** (рисунок 2.1).

Откроется диалоговое окно **Новая таблица** (рисунок 2.2).

В этом окне можно выбрать один из пяти возможных способов создания таблицы:

- Режим таблицы (Datasheet View) – таблица создается путем ввода имен полей в заголовок каждого столбца.

- Конструктор (Design View) – таблица создается путем составления списка имен полей и задания свойств каждого поля.

- Мастер таблиц (Table Wizard) – таблица создается автоматически с помощью программы-мастера, которая предлагает выбрать поля из списка и содержит заголовки для различных видов таблиц (свыше 100).

- Импорт таблиц (Import Table) – таблица создается путем импорта данных из другой базы или из другой электронной таблицы.

– Связь с таблицами (Link Table) – таблица создается путем установления связей с таблицей, существующей в другой базе данных.

Если речь идет о создании новой таблицы, то удобнее всего воспользоваться для этого **режимом Конструктора (Design View)**. Этот способ удобен тем, что позволяет одновременно задать поля будущей таблицы и назначить свойства этих полей.

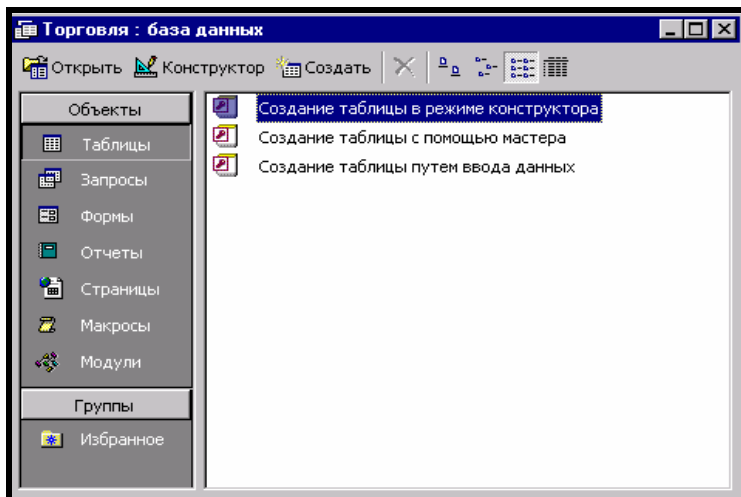


Рисунок 2.1 – Окно базы данных

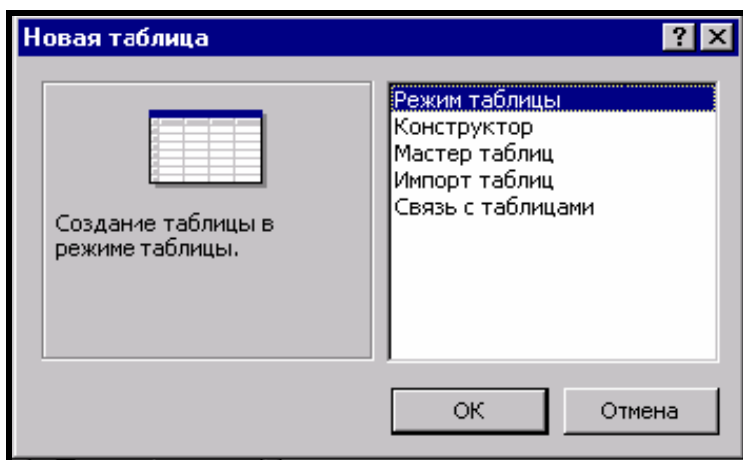


Рисунок 2.2 – Создание таблицы

2.1.1 Создание таблицы в режиме Конструктора

Для создания таблицы в режиме **Конструктора** необходимо выбрать строку **Конструктор** в окне **Новая таблица** и нажать кнопку **ОК**. Откроеется окно конструктора таблицы (рисунок 2.3).

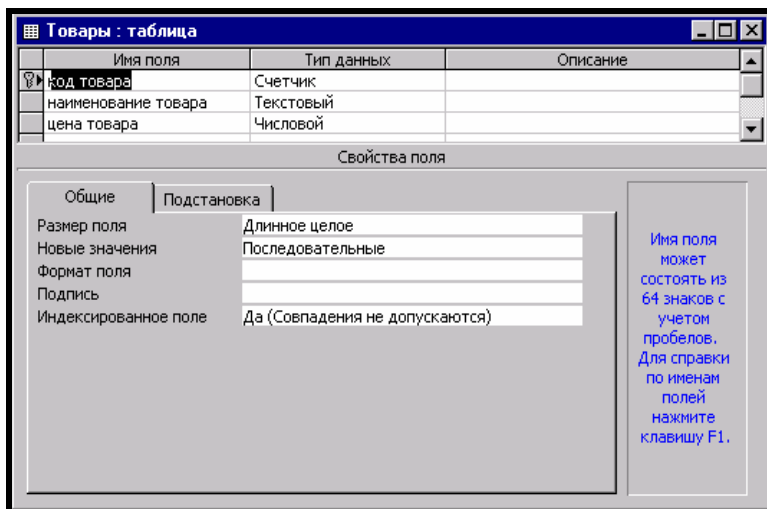


Рисунок 2.3 – Создание таблицы в режиме Конструктора

Окно Конструктора таблицы имеет следующую структуру:

- столбец маркировки (черный треугольник в этом столбце указывает на ту строку, в которой в данный момент находится курсор ввода; значок ключа – на то, что данное поле является ключевым);
- столбец – имя поля;
- столбец – тип данных;
- столбец – описание.

В нижней левой части окна Конструктора представлены свойства полей.

В нижней правой части окна Конструктора отображается справочная информация о допустимых на каждом этапе проектирования действиях.

2.2 Определение имен полей

Имя поля может иметь длину до 64 символов и содержать пробелы и любые специальные символы, кроме точек, восклицательных знаков и прямых скобок. В таблице не должно быть двух полей с одинаковыми именами, поскольку Access идентифицирует поле по его имени.

2.3 Типы данных

Тип данных указывает, какие данные могут быть введены. Access позволяет выбрать один из следующих типов данных:

Текстовый – тип данных, используемый для хранения обычного неформатированного текста ограниченного размера (до 255 символов).

Поле Мемо – специальный тип данных для хранения больших объемов текста (до 65 535 символов). Физически текст не хранится в поле. Он хранится в другом месте базы данных, а в поле хранится указатель на него, но для пользователя такое разделение заметно не всегда.

Числовой – тип данных для хранения действительных чисел.

Дата/время – тип данных для хранения календарных дат и текущего времени.

Денежный – тип данных для хранения денежных сумм.

Счетчик – специальный тип данных для уникальных (не повторяющихся в поле) натуральных чисел с автоматическим наращиванием. Естественное использование — для порядковой нумерации записей.

Логический – тип для хранения логических данных (могут принимать только два значения, например Да или Нет).

Поле объекта OLE – специальный тип данных, предназначенный для хранения объектов OLE, например мультимедийных. Реально, конечно, такие объекты в таблице не хранятся. Как и в случае полей MEMO, они хранятся в другом месте внутренней структуры файла базы данных, а в таблице хранятся только указатели на них (иначе работа с таблицами была бы чрезвычайно замедленной).

Гиперссылка – специальное поле для хранения адресов URL Web-объектов Интернета. При щелчке на ссылке автоматически происходит запуск браузера и воспроизведение объекта в его окне.

Мастер подстановок – это не специальный тип данных. Это объект, настройкой которого можно автоматизировать ввод в данных поле так, чтобы не вводить их вручную, а выбирать из раскрывающегося списка.

По умолчанию устанавливается тип данных Текстовый. Чтобы изменить тип данных, надо щелкнуть мышью на строке поля в столбце Тип данных (Data Type), после чего щелкнуть на кнопке раскрывающийся список и выбрать из раскрывающегося списка нужный тип данных (рисунок 2.4).

2.4 Заполнение полей таблицы в окне конструктора

Для каждого поля в окне конструктора последовательно вводятся имя, затем тип данных. Если назначение поля неочевидно по его имени, необходимо ввести произвольное описание поля в столбец «*Описание*». Переход к следующей ячейке строки осуществляется с помощью клавиши TAB или

щелчком мыши на этой ячейке. Переход к заполнению следующего поля осуществляется клавишами TAB, ENTER или щелчком мыши.

2.5 Свойства полей

Каждое поле имеет свой набор свойств, от которых зависит то, как содержимое этого поля отображается на экране, как оно хранится в базе данных, как оно проверяется и как им можно управлять. Некоторые свойства являются общими для всех типов полей, другие относятся исключительно к отдельным типам данных.

Некоторые из свойств задаются по умолчанию. Свойства полей не являются обязательными. Их можно настраивать по желанию, а можно и не трогать.

Все типы полей, кроме полей счетчика и объекта OLE, имеют следующие основные свойства:

- Формат поля (Format) – определяет, как должно отображаться содержимое поля.
- Подпись (Caption) – определяет тот текст, который будет отображаться рядом с полем или в отчете.
- Значение по умолчанию (Default Value) – определяет значение, которое вводится в поле автоматически при добавлении новых записей.
- Условие на значение (Validation Rule) – определяет область или диапазон значений данных, допустимых в поле.
- Сообщение об ошибке (Validation Text) – задает текст сообщения, которое появляется в строке состояния при попытке ввести в поле недопустимое значение.
- Обязательное поле (Required) – содержит значение Да (Yes) или Нет (No), в зависимости от того, обязательно ли должно это поле иметь какую-либо запись.

Следующие свойства относятся в основном к текстовым полям:

- Размер поля (Field Size) – максимально допустимое количество символов для данного поля.
- Маска ввода (Input Mask) – обеспечивает ввод данных в указанном формате.
- Индексированное поле (Indexed) – указывает Access, должны ли создаваться индексы для этого поля: индексирование поля ускоряет процесс сортировки, поиска и фильтрации содержимого. Поле, являющееся первичным ключом таблицы, индексировано всегда. Стоит задавать индексацию тех полей, по которым возможно проведение сортировки в формах или отчетах.
- Сжатие Unicode (Unicode Compression) – представляет собой схему кодировки всех известных символов, для достижения чего вместо одного

байта каждый символ представлен двумя. Когда эта функция отключена, Access компрессирует все символы, начинающиеся с байта 0, при сохранении и распаковывает их при обращении к ним.

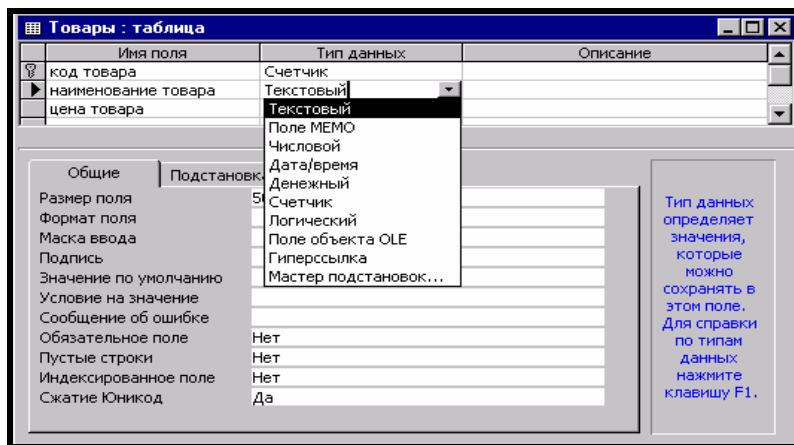


Рисунок 2.4 – Выбор типа данных

Поля числового и денежного типов имеют одно дополнительное свойство – число десятичных знаков, которое определяет, какое количество десятичных знаков после запятой будет отображаться на экране и храниться в поле.

Чтобы установить свойство поля, надо выделить строку поля, после чего в левой нижней части окна конструктора откроется список свойств данного поля. Щелчком мыши установить курсор на строку свойства и выбрать нужное значение.

2.5.1 Задание маски ввода

При вводе данных, например в поле дата поступления, было бы удобно, чтобы в этом поле заранее вводилась точка в качестве разделителя – тогда ее не надо было вводить вручную.

Форматы для ввода данных называются масками ввода.

В состав Access входит мастер ввода масок. Чтобы задать маску ввода, надо:

- 1 Щелкнуть по строке свойства – Маска ввода, появится кнопка с тремя точками.
- 2 Щелкнуть по кнопке с тремя точками. Пользователю будет предложено сохранить таблицу, после чего откроется окно **Создание масок ввода**.

3 Выбрать тип маски ввода (например, краткий формат даты для поля дата поступления) и щелкнуть по кнопке **Готово** (Finish). Диалоговое окно **Создание масок ввода** закроется, и маска ввода будет внесена в свойства поля.

Хранение данных в базе вместе с маской ввода увеличивает размер базы данных, но позволяет облегчить ввод данных.

2.5.2 Задание свойства условие на значение

Условие на значение (Validation Rule) – это свойство, с помощью которого можно проверять данные, вводимые в таблицу или форму.

Условие на значение для поля устанавливается в строке свойства **Условие на значение**. Когда после ввода данных пользователь переключается на следующее поле или щелкнет вне его, то Access проверяет, соответствуют ли введенные данные установленному условию на значение. Если условие не выполняется, выводится **Сообщение об ошибке**.

Чтобы задать условие на значение для поля, надо:

- 1 Открыть таблицу в режиме конструктора.
- 2 Выделить поле.
- 3 Щелкнуть в строке свойства **Условие на значение** и ввести условие с помощью логических операторов (например, для поля цена введем <50 000).
- 4 В строке свойства **Сообщение об ошибке** ввести текст сообщения, которое должно отображаться в строке состояния при вводе недопустимых данных.

2.6 Выбор ключевого поля

В реляционной базе данных таблицы должны быть связаны друг с другом. Эта связь устанавливается с помощью уникальных полей. Уникальные поля – это такие поля, в которых значения не могут повторяться. Например, в базе данных **Торговля** в таблице1 таким уникальным полем может быть №магазина. Такое поле (или комбинация полей), которое однозначно идентифицирует запись в таблице, называется первичным ключом.

Если в таблице нет такого поля, которое могло бы служить ключом, можно объединить несколько полей. Чтобы создать сложный ключ, состоящий из нескольких полей, надо, удерживая нажатой клавишу CTRL, выделить те поля, которые необходимо включить в ключ и щелкнуть правой кнопкой мыши, в открывшемся контекстном меню выбрать пункт **Ключевое поле**.

Если никакая комбинация полей не подходит для ключа, можно ввести в таблицу дополнительное поле – например, порядковый номер записи, присвоив ему тип – счетчик.

Чтобы задать ключевое поле, надо:

1 Выделить поле (или поля), выбранное в качестве ключевого, щелчком мыши слева от столбца **Имя поля**.

2 Нажать кнопку **Ключевое поле** на панели инструментов или щелкнуть правой кнопкой мыши, и в открывшемся контекстном меню выбрать пункт **Ключевое поле**.

Если к моменту выхода из режима конструктора первичный ключ для создаваемой таблицы объявлен не будет, Access выдаст запрос о необходимости включения в таблицу поля первичного ключа. Необходимо выбрать нужный ответ Да или Нет.

2.6.1 Свойства ключевого поля

Для ключевого поля Access автоматически устанавливает в строке свойства **Обязательное поле** значение Да(Yes), а в строке свойства **Индексированное поле** – значение Да (совпадение не допускается). Если в качестве первичного ключа служит группа из нескольких полей, то все поля объявляются обязательными, но совпадения внутри каждого из них допускаются.

2.7 Сохранение таблицы

При закрытии окна конструктора таблицы Access напомнит, что созданную структуру таблицы надо сохранить. Для сохранения структуры выберите пункт **Да**. При выборе пункта **Нет** вы останетесь в режиме конструктора.

2.8 Присвоение имени объектам

Access позволяет называть объекты баз данных так, как вам нравится, но есть неофициальное соглашение Лежински-Реддика, следуя которому можно сделать имена объектов понятными и для других людей. По соглашению, имя объекта начинается тегом, идентифицирующим тип объекта: тбл(tbl), фрм(frm), отч(rpt), зпр qry) и др. В именах не используются пробелы и знаки препинания, первая буква имени должна быть прописной, например тблМагазины, тблПоставки.

2.9 Межтабличные связи

Access позволяет строить реляционные базы данных, отдельные таблицы которых могут быть связаны между собой. Связь между таблицами определяет тип отношения между их полями. Как правило, связывают ключевое поле одной таблицы с соответствующим ему полем другой таблицы, которое называется полем внешнего ключа (рисунок 2.5).

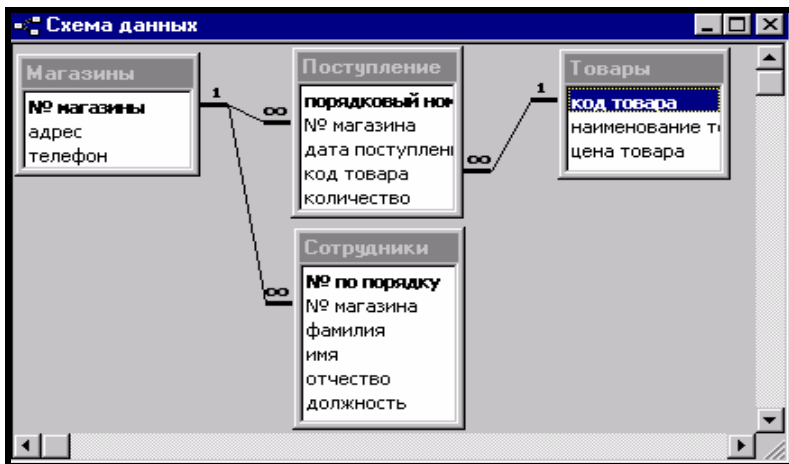


Рисунок 2.5 – Схема данных

При наличии связей между таблицами Access автоматически будет выбирать связанные данные из таблиц в отчетах, запросах, формах.

Существуют три типа связей:

- один-к-одному – каждая запись таблицы А не может быть связана более чем с одной записью таблицы Б;
- один-ко-многим – одна запись в таблице А может быть связана со многими записями таблицы Б;
- многие-ко-многим – каждая запись в таблице А может быть связана со многими записями в таблице Б, а каждая запись в таблице Б – со многими записями в таблице А.

В Access существует возможность отображения связанных таблиц в окне главной таблицы. Для этого необходимо активизировать главную таблицу в режиме просмотра. Между областью выделения и первым столбцом таблицы имеется узкий столбец, в котором против каждой записи проставлен знак плюс (+) (рисунок 2.6). При щелчке на нём левой кнопкой мыши отображается часть подчинённой таблицы, где содержатся только те записи, которые связаны с текущей записью главной таблицы. При этом плюс (+) меняется на минус (-).

2.9.1 Создание схемы данных

Чтобы создать связи между таблицами, надо:

1 В меню **Сервис** выбрать команду **Схема данных**. На экране появится окно **Схема данных**, в котором можно просмотреть и определить связи между таблицами базы данных, если они установлены.

Товары : таблица				
	код товара	наименовани	цена товара	
	1	сметана	980	
	порядковый	№ магазина	дата поступления	количество
	2	1	01.02.03	125000
	4	2	03.02.03	1260
		(Счетчик)	0	0
+	2	сыр	5600	
+	3	молоко	500	
+	4	кефир	4500	
*		(Счетчик)	0	

Рисунок 2.6 – Связанные таблицы в режиме просмотра

2 Если связи между таблицами не установлены, окно **Схема данных** будет пустым и необходимо добавить в него таблицы, для которых устанавливаются связи.

3 Чтобы добавить в окно **Схема данных** таблицы, можно щелкнуть правой кнопкой мыши на свободном месте окна и в контекстном меню выбрать пункт **Добавление таблицы**. Откроется окно **Добавление таблицы**. Или щелкнуть по кнопке **Отобразить связи** на панели инструментов и выбрать из списка нужные таблицы.

4 В окне **Добавление таблицы** выделить название таблицы и нажать кнопку **Добавить**. Так необходимо добавить все таблицы, между которыми надо установить связи.

5 Перетащить с помощью мыши используемое для связи поле первичного ключа главной таблицы к соответствующему полю связанной таблицы. На экране появится окно **Изменение связей**. В этом окне главная таблица и первичный ключ изображаются в левом столбце, а подчиненная таблица и внешний ключ – в правом столбце. Установить флажки: Обеспечение целостности данных, Каскадное обновление связанных полей, Каскадное удаление связанных записей.

6 В нижней части окна будет предложен тип связи между таблицами.

7 Нажать кнопку **ОК**.

2.9.2 Задание условий целостности

Условие целостности предназначено для того, чтобы обеспечить соответствие записей в подчиненной таблице записям в главной таблице. Оно

предотвратит непреднамеренное удаление пользователем тех записей в главной таблице, которые связаны с записями в подчиненной таблице.

Если допускается каскадное обновление связанных полей, то при изменении значений первичного ключа в главной таблице происходит автоматическое обновление значений внешнего ключа в подчиненной таблице.

Если разрешено каскадное удаление связанных записей, то удаление записи в главной таблице приводит к удалению всех связанных записей в подчиненной таблице.

Для задания условий целостности необходимо:

– Выбрать в пункте меню **Сервис** пункт **Схема данных**. Откроется окно **Схема данных**.

– Двойным щелчком мыши на существующих линиях связи открыть диалоговое окно **Изменение связей** (рисунок 2.7.) и в нем установить флажки: Обеспечение целостности данных, Каскадное обновление связанных полей, Каскадное удаление связанных записей.

– Щелкнуть по кнопке **ОК**.

– Закрыть окно **Схема данных**. Сохранить внесенные изменения.

2.10 Ввод данных в таблицу

После задания всех полей и определения их свойств выполняется ввод данных в таблицу.

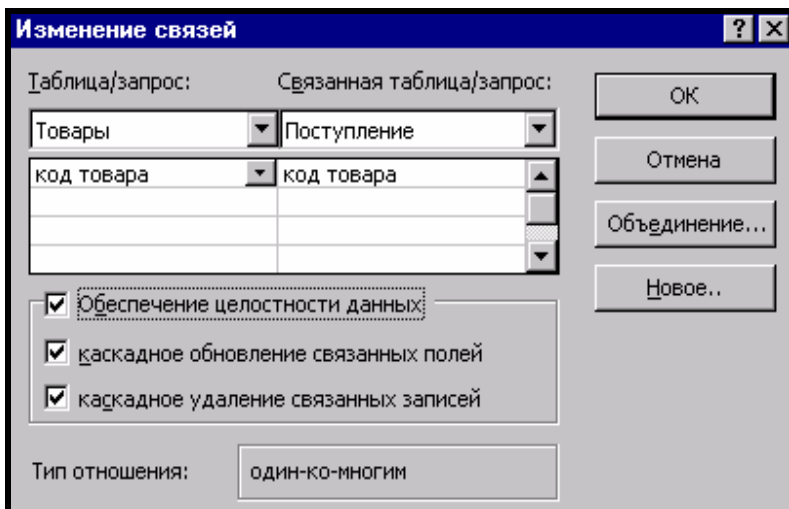


Рисунок 2.7 – Задание условий целостности

Чтобы заполнить таблицу, следует:

1 Выделить имя таблицы в окне базы данных и нажать кнопку Открыть. Если в данный момент открыто окно конструктора таблицы, необходимо перейти в режим отображения таблицы, нажав кнопку **Вид** на панели инструментов. При этом вид кнопки изменится, и ее повторное нажатие приведет к возврату в окно конструктора таблицы.

2 Ввести информацию в первое поле, щелкнув в нем мышью. Для перехода к следующему полю нажмите клавишу TAB или ENTER. Затем можно продолжить ввод информации с нажатием клавиши TAB после заполнения каждой ячейки, пока не дойдете до конца записи. Для перехода на следующую запись необходимо нажать клавиши TAB или ENTER в конце строки.

3 Чтобы изменить ширину столбца, установите указатель мыши на разделительной линии между заголовками столбцов. Когда указатель мыши изменит свою форму и примет вид двунаправленной стрелки, перетащите его в нужном направлении.

На листе данных активная запись обозначается треугольным маркером, а пустая запись – звездочкой. Для обозначения записи, в которой выполняется ввод, используется изображение карандаша. Все маркеры появляются в столбце маркировки, расположенном в левой части листа данных (рисунок 2.8).

Поле счетчика (Счетчик) заполняется автоматически. Access самостоятельно увеличивает значение переменной в этом поле, присваивая каждой записи новый номер.

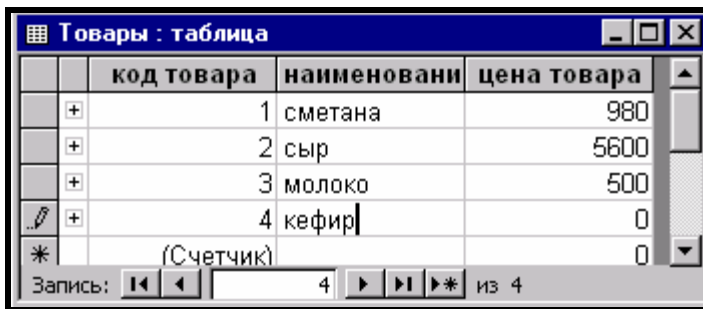


Рисунок 2.8 – Окно таблицы в режиме заполнения

2.11 Перемещение по таблице

В программе Access существуют различные способы перемещения по таблице.

Для активизации одного поля записи достаточно выполнить щелчок на нем. Поле, которое находится в невидимой части таблицы, можно отобразить на экране с помощью полос прокрутки.

Перемещаться по таблице в любом направлении позволяют клавиши управления курсором. С помощью клавиш TAB и ENTER можно перемещаться по полям слева направо, а посредством комбинации клавиш SHIFT+TAB – в обратном направлении.

Наиболее удобным средством перемещения по таблице являются кнопки из области **Запись**, расположенной в левом нижнем углу листа данных. С их помощью можно перейти к любому блоку данных. Эти кнопки соответствуют командам из подменю **Перейти** меню **Правка**. Для активизации записей с известным номером достаточно ввести этот номер в специальное **Поле номера записи** и нажать клавишу ENTER.

В программе Access по умолчанию содержимое поля полностью выделяется, если переход произведен с помощью клавиш управления курсором или клавиши TAB. Если же поле активизировано путем выполнения на нем щелчка, то в нем появится мерцающий курсор ввода, свидетельствующий о том, что можно начинать ввод.

Программа Access автоматически сохраняет каждую запись по завершении ее обработки.

2.12 Редактирование данных в таблице

Для редактирования данных в таблице необходимо открыть таблицу, установить указатель мыши щелчком на той ячейке, содержимое которой надо изменить. Указатель примет форму “I”. Внести изменения, нажать ENTER.

Двойным щелчком можно выделить слово и затем набрать другое слово, которое его заменит.

Для того чтобы выделить все содержимое поля, установите указатель мыши на левом краю поля. При этом указатель изменит форму. Выделить все поле в этот момент можно одним щелчком.

В Access существуют два режима обработки данных таблицы: режим замены и режим вставки.

В режиме замены содержимое активизированного поля полностью маркируется и удаляется в момент ввода новых данных.

В режиме вставки внутри активизированного поля появляется курсор ввода. При вводе новых данных старые не удаляются, а смещаются вправо. Ввод осуществляется в той позиции, в которой был выполнен щелчок.

Переход из режима вставки в режим замены и наоборот позволяет клавиша INS. Режим замены обозначается индикатором ЗМЩ в строке состояния окна Access.

2.13 Изменение структуры таблицы

Изменение структуры таблицы (типа данных полей, порядка их следования и т.д.) выполняется в режиме конструктора таблицы. Изменения в структуре записи заполненной таблицы чреваты потерей данных.

2.14 Сжатие данных

Access может автоматически сжимать базу данных при ее закрытии.

Для активизации функции сжатия данных:

- в пункте меню **Сервис** выбрать команду **Параметры** и вкладку **Общие**;
- проверить, установлен ли флажок «Сжимать» при закрытии;
- щелкнуть на кнопку **ОК**.

2.15 Внешний вид таблицы

Access позволяет изменить макет таблицы.

Макет таблицы – это вид, в котором данные представлены в окне режима таблицы.

Макет таблицы можно изменить для всех таблиц данного приложения - меню **Сервис** команда **Параметры** или только для конкретной таблицы.

Для установки режима макета таблицы в пункте меню **Формат** надо выбрать команду **Режим таблицы**. Откроется окно **Формат таблицы**. Здесь можно сделать необходимые установки.

Чтобы установить вид, начертание и размер шрифта для записей таблицы, надо выбрать в пункте меню **Формат** команду **Шрифт**.

Порядок выполнения работы

1 На чистом листе бумаги спроектировать базу данных «**Торговля**»: распределить поля по четырем таблицам «Магазины», «Сотрудники», «Товары», «Поступление»; указать типы полей, ключи и связи.

2 Выполните запуск MS Access следующим образом:

ПУСК – ПРОГРАММЫ – MS Access

3 После запуска программы на экране откроется окно, в котором можно выбрать один из следующих вариантов продолжения работы:

- создание новой базы данных;
- создание базы данных с помощью мастера;

– открытие одной из существующих баз данных.

4 Для создания новой базы данных «*Торговля*» выберите переключатель «*Новая база данных*» и нажмите **ОК**. Откроется диалоговое окно «Файл новой базы данных», в котором можно выбрать имя файла и место его расположения. Сохранить файл базы данных надо до того, как приступите к созданию таблиц. Таким образом, создается пустая база данных, в которую впоследствии можно поместить таблицы, формы, запросы, отчеты.

Файл базы данных сохранить на диске Z:, в личной папке своей группы с именем «Торговля». **Например:** Z:\GB12\ИвановаИ.И.\Торговля

5 В режиме «*Конструктор*» создать четыре таблицы для базы данных «*Торговля*»:

Таблица 1 – «Магазины» содержит следующие поля:

- № магазина – **ключ** (тип данных – счетчик);
- адрес (тип данных – текстовый);
- телефон (тип данных – текстовый).

Таблица 2 – «Товары» содержит следующие поля:

- код товара – **ключ** (тип данных – счетчик);
- наименование товара (тип данных – текстовый);
- цена (тип данных – числовой).

Таблица 3 – «Поступление» содержит следующие поля:

- порядковый номер – **ключ** (тип данных – счетчик);
- № магазина (тип данных – числовой);
- дата поступления (тип данных – дата\время);
- код товара (тип данных – числовой);
- количество (тип данных – числовой).

Таблица 4 – «Сотрудники» содержит следующие поля:

- № п\п **ключ** (тип поля – счётчик);
- № магазина (тип поля – числовой);
- фамилия (тип поля – текстовый);
- имя (тип поля – текстовый);
- отчество (тип поля – текстовый),
- должность (тип поля – текстовый).

6 При описании поля – дата поступления – задать **маску ввода**.

7 Для поля цена – ввести свойство «Условие на значение» (<50 000). В строке свойства «Сообщение об ошибке» ввести текст сообщения «Проверить цену. Цена должна быть <50 000».

8 Для каждой таблицы задать **ключевое поле**.

9 Установить связи между таблицами:

– между таблицей 1 и таблицей 4 – по полю «№ магазина» (один - ко - многим);

– между таблицей 2 и таблицей 3 – по полю «код товара» (один - ко - многим);

– между таблицей 1 и таблицей 3 – по полю «№ магазина» (один - ко - многим);

10 Установить флажок обеспечения целостности данных. Установить флажок каскадного обновления связанных полей и каскадного удаления связанных записей или без них.

11 Заполнить базу данных. Группу товаров выбрать из таблицы 2.1 по номеру студента в журнале.

В таблице 1 должно быть не менее 5 записей; в таблице 2 – не менее 10 записей (допускается одинаковое наименование товаров с разной ценой (разные коды товаров)), в таблице 3 – не менее 30 записей (предусмотреть поступление товаров в один магазин за разные даты и разных товаров в один магазин), в таблице 4 предусмотреть в одном магазине несколько сотрудников (зав. магазином, зав. отделом, продавец I кат. и т.д.).

Таблица 2.1 – Наименование группы товаров

№ п/п	Наименование группы товаров
1	Вычислительная техника
2	Канцтовары
3	Одежда
4	Обувь
5	Мебель
6	Косметика
7	Ткани
8	Детские товары
9	Рыба
10	Хлебобулочные изделия
11	Полуфабрикаты
12	Кондитерские изделия
13	Овощи
14	Молочные продукты

КОНТРОЛЬНЫЕ ВОПРОСЫ

- 1 Проектирование базы данных. Создание базы данных.
- 2 Создание таблицы. Режим Конструктора.
- 3 Описание полей базы данных.
- 4 Тип данных.
- 5 Свойства полей.
- 6 Задание маски ввода.

- 7 Выбор ключевого поля.
- 8 Непосредственный ввод данных в таблицу.
- 9 Редактирование данных в таблице.
- 10 Удаление записи из таблицы.
- 11 Взаимосвязи таблиц. Создание схемы данных.
- 12 Обеспечение целостности данных .

Лабораторная работа № 2

СОЗДАНИЕ ФОРМ

Цель работы: создание формы. Мастер и конструктор форм. Создание многотабличной формы.

1 ФОРМЫ

1.1 Создание формы

Форма – объект базы данных, предназначенный для ввода данных в таблицу, кнопочная форма – для открытия других форм.

Перед созданием формы необходимо создать структуру таблицы или запроса, для которых будет создаваться форма.

Формы обеспечивают пользователям возможность ввода данных без необходимости знать, как устроена таблица. Одна форма может содержать данные из многих таблиц, благодаря чему обеспечивается возможность ввода данных в различные таблицы из одного места. Макет формы можно заполнить таким образом, что форма на экране будет выглядеть точно так же, как и первоисточник: заявление о приеме, квитанция, заполняемая клиентами, и прочие документы, используемые для сбора данных, подлежащих вводу в базу данных.

Для создания формы необходимо:

- 1 На вкладке **Формы** окна базы данных нажать кнопку **Создать**.
- 2 На экране появится диалоговое окно **Новая форма**, в котором программа предлагает выбрать способ создания формы (рисунок 1.1).

Существует несколько способов создания форм:

- 1 Автоформа (Auto Form) – автоматическое создание форм.
- 2 Мастер форм (Form Wizard) – создание форм с помощью мастера.
- 3 Конструктор (Form Design View) – создание формы вручную в режиме конструктора.
- 4 Мастер диаграмм – создание формы с диаграммой на основе выбранных полей таблицы.

5 Мастер сводных таблиц – создание сводной таблицы Microsoft Excel на основе таблиц и запросов Access.

1.1.1 Создание автоформы

Создание автоформы используется в том случае, когда пользователю необходимы формы, которые содержат всю информацию из отдельной таблицы или запроса. В автоформе каждое поле расположено на отдельной строке, с левой стороны которой отображается надпись к данному полю.

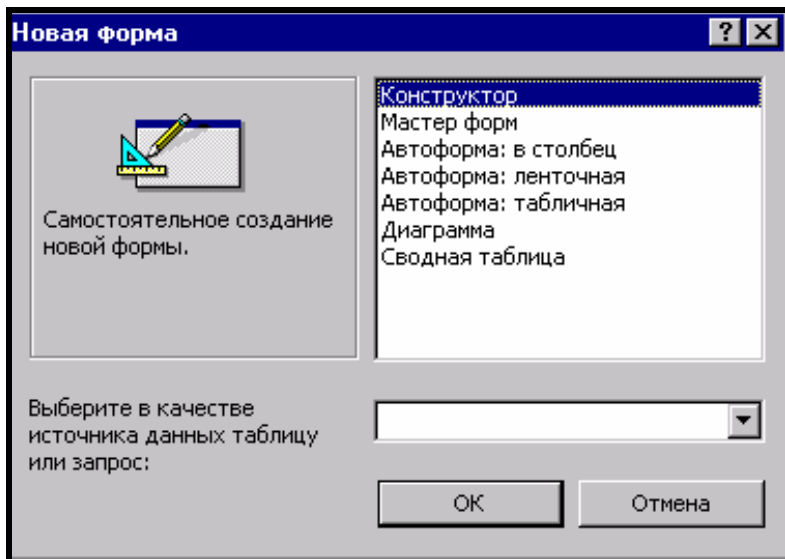


Рисунок 1.1 – Создание формы

Автоформы бывают трех типов:

1 Автоформа: в столбец – для каждой записи отводится отдельная страница формы. Данная форма подходит для записей с большим числом полей. Для перехода от одной записи к другой необходимо щелкнуть на поле номера записи.

2 Автоформа: ленточная – каждая запись размещается в одной строке таблицы. Данная форма позволяет наглядно представить несколько записей.

3 Автоформа: табличная – имеет вид таблицы и выступает в качестве подчиненной формы в составных формах.

Для создания автоформы необходимо:

- 1 На вкладке **Формы** окна базы данных нажать кнопку **Создать**.
- 2 Выбрать способ создания формы – автоформа.

3 В нижней части окна **Новая форма** из списка выбрать таблицу или запрос, на основе которого будет создана форма; нажать кнопку **ОК**.

4 На экране откроется готовая форма. При ее закрытии Access предложит сохранить форму, указав ее имя.

1.2 Ввод и редактирование данных в форме

В табличных формах информация вводится и редактируется так же, как это выполняется в таблицах в режиме просмотра. Поле для редактирования выбирается установкой указателя мыши на левом краю поля с последующим щелчком.

В ленточных формах и в формах в столбец данные, подлежащие редактированию, выделяются перетаскиванием мыши, после чего вводятся правильные данные.

Переход из поля в поле в форме осуществляется клавишей **Tab** или клавишами управления курсором. Для перехода к следующей записи надо нажать клавишу **Tab**, находясь на последнем элементе управления формы.

Во всех формах для перехода между записями можно использовать **кнопки перехода**.

Для удаления записи из формы ее необходимо маркировать, щелкнув в столбце маркировки (выделение записи отменяется при выполнении щелчка вне столбца маркировки), и нажать клавишу **Del**.

1.3 Мастер форм

Использование мастера форм позволяет включать в форму поля из различных таблиц.

Для создания формы с помощью мастера необходимо:

1 Открыть вкладку **Формы** окна базы данных и нажать кнопку **Создать**.

2 В появившемся окне **Новая форма** задать создание формы с помощью мастера и нажать кнопку **ОК** (рисунок 1.2).

3 В **первом** диалоговом окне мастера (рисунок 1.3) выбрать в списке **Таблицы\запросы** таблицу, из которой в форму будут включены поля, а в списке **Допустимые поля** выбрать поля для подготавливаемой формы. С помощью кнопок **>** или **>>** перенести выбранные поля в список **Выбранные поля**.

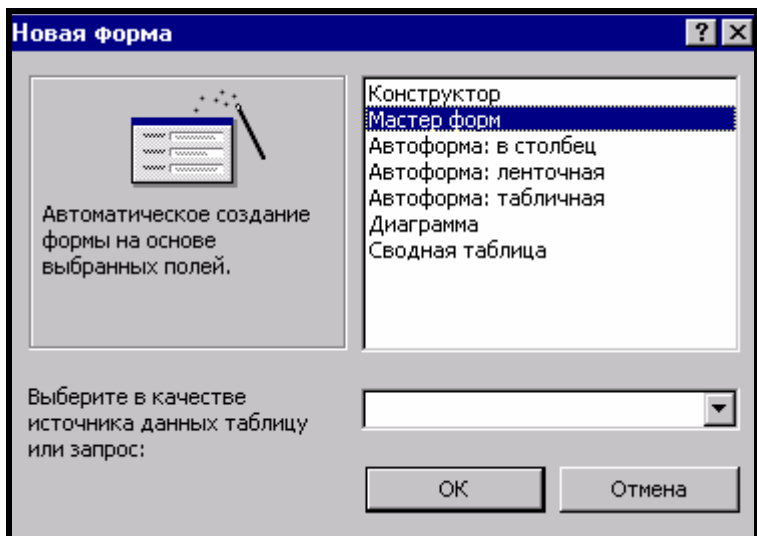


Рисунок 1.2 – Создание формы с помощью мастера

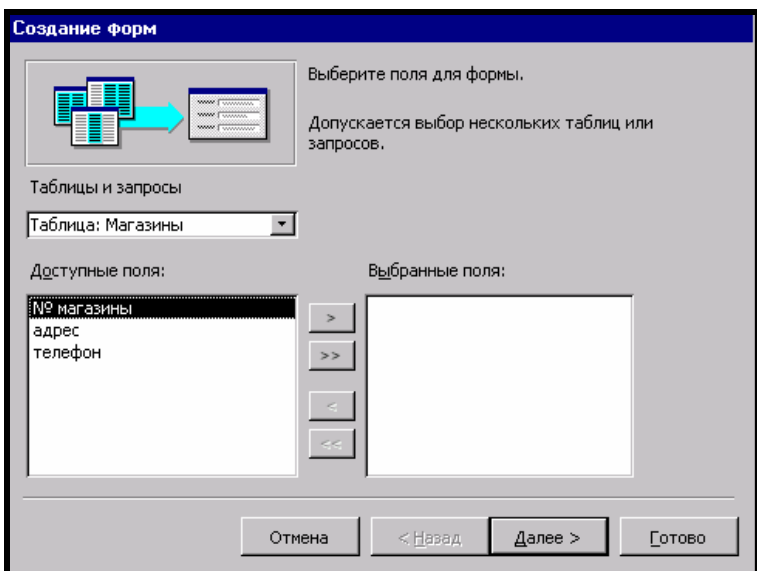


Рисунок 1.3 – Выбор полей

4 В списке *Таблицы\запросы* последовательно можно выбирать таблицы, из которых необходимо включать поля в форму. При включении в форму полей из различных таблиц будет создана многотабличная форма.

5 После отбора всех полей из таблиц в форму нажать кнопку *Далее* для перехода в следующее окно мастера форм.

6 Во *втором* окне мастера форм необходимо выбрать *вид* формы. Для этого активизируйте один из переключателей (в один столбец; ленточный; табличный; выровненный) и нажмите кнопку *Далее*.

7 В *третьем* окне мастера задать *стиль* оформления и нажать кнопку *Далее*.

8 В последнем окне мастера ввести *имя формы* и выбрать одну из опций: открыть форму для просмотра и ввода данных или изменить макет формы. Нажать кнопку *Готово*.

1.3.1 Создание подчиненной формы

Подчиненная форма – это вложенная форма, позволяющая отображать данные из связанной таблицы в тех случаях, когда одной записи из главной таблицы может соответствовать несколько записей подчиненной таблицы (связь один-ко-многим). Подчиненная форма внедряется в главную связь (один-ко-многим). Если главная форма уже достаточно сложна, то можно выбрать отображение подчиненной формы не как вложенной, а как связанной. В этом случае в главной форме содержится кнопка, щелчком на которой открывается подчиненная форма.

Чтобы для таблиц, связанных один-ко-многим, создать форму, необходимо:

1 Открыть вкладку *Формы* окна базы данных и нажать кнопку *Создать*.

2 Запустить мастер форм.

3 В списке *Таблицы\запросы* выбрать основную таблицу и скопировать нужные поля в список *Выбранные поля*.

4 В списке *Таблицы\запросы* выбрать подчиненную таблицу и скопировать нужные поля в список *Выбранные поля*; нажать кнопку *Далее*.

5 Выбрать вид представления данных: хотите ли вы, чтобы форма основывалась на главной или на связанной форме, при этом в качестве исходной можно выбрать главную или подчиненную таблицу (рисунок 1.4).

6 Опции *Подчиненные формы* и *Связанные формы* определяют отношения между таблицами. Выбрав опцию *Подчиненные формы*, создают главную форму, содержащую подчиненную форму. Если выбрать параметр *Связанные формы*, то в главной форме будет содержаться кнопка, щелчком на которой открывается подчиненная форма. Нажать кнопку *Далее*.

7 Выбрать *внешний вид* подчиненной формы. Нажать кнопку *Далее*.

8 Выбрать *стиль оформления* формы. Нажать кнопку *Далее*.

9 Ввести *заголовок главной формы*. Для подчиненной формы оставить предложенное программой имя.

10 Установить одну из опций: открыть форму для просмотра (рисунок 1.5) или ввода данных или изменить макет формы (рисунок 1.6). При выборе второй опции и нажатии кнопки *Готово* на экране появится проект формы.

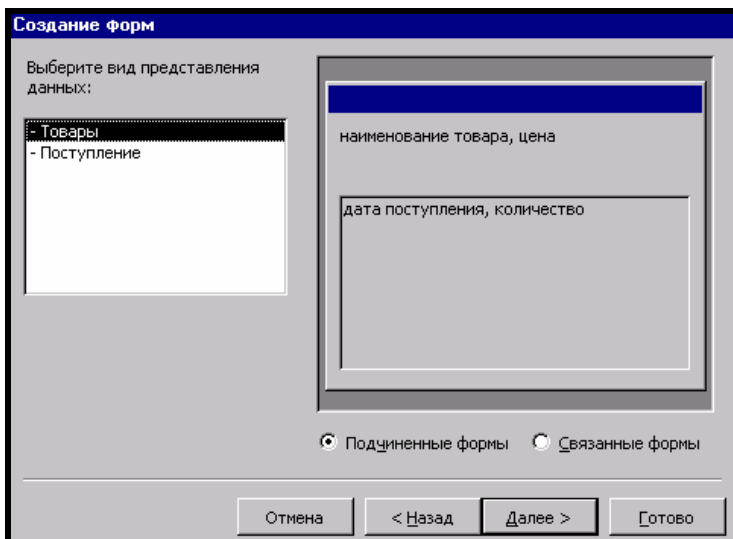


Рисунок 1.4 – Создание подчинённой формы

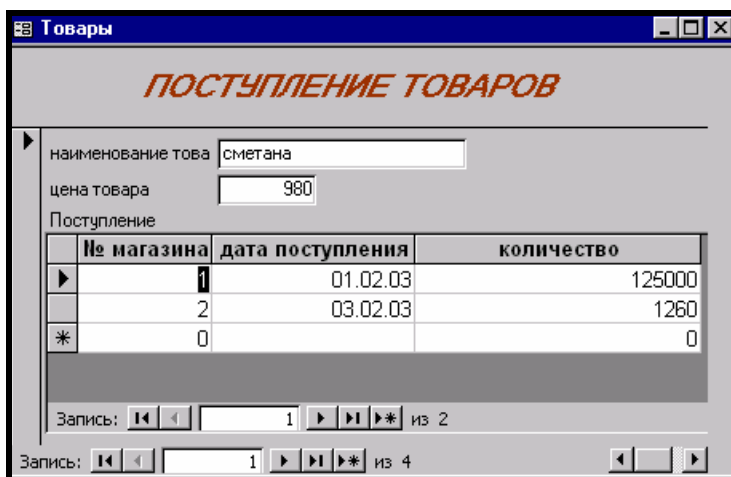


Рисунок 1.5 – Макет подчиненной формы для ввода данных

1.4 Конструктор форм

Режим конструктора используется для создания форм вручную.

Для создания формы в режиме конструктора необходимо выполнить следующие действия:

- 1 В окне базы данных выбрать вкладку **Формы**.
- 2 Нажать кнопку **Создать**.
- 3 В диалоговом окне **Новая форма** выбрать пункт **Конструктор**.
- 4 Выбрать имя таблицы или запроса, содержащих данные, на основе которых будет создана форма. Если форма не будет содержать данных (например, если форма будет использоваться как кнопочная для открытия других форм или отчетов или как специальное диалоговое окно), ничего не выбирайте в списке.

Примечание. Форма, использующая данные из нескольких таблиц, должна быть основана на запросе, включающем эти таблицы.

- 5 Перетащить поля из таблицы в область данных.
- 6 Сохранить таблицу, указав имя.
- 7 Изменения в форму можно внести в режиме Конструктора.

Чтобы открыть форму в режиме конструктора, необходимо нажать соответствующую кнопку на панели инструментов.

Окно конструктора содержит горизонтальные и вертикальные линейки, панель форматирования и панель инструментов. Дополнительно можно открыть панель элементов, с помощью которой в форму можно ввести новые объекты (см. рисунок 1.5)

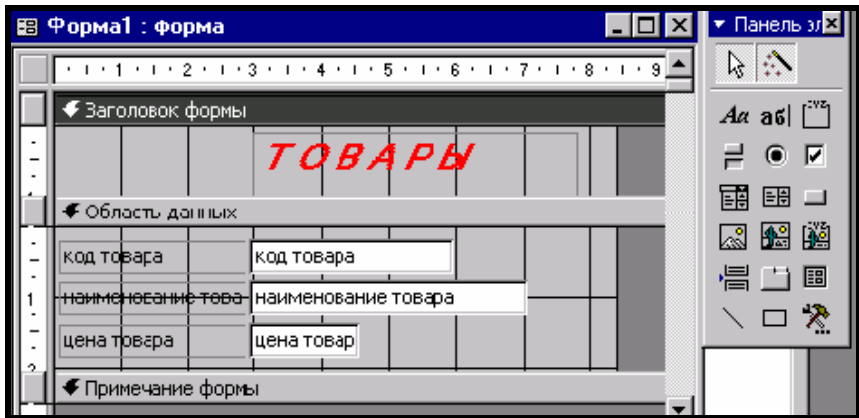


Рисунок 1.6 – Вид формы в конструкторе

Форма в режиме конструктора имеет три раздела и несколько различных элементов управления (см. рисунок 1.6):

1 **Заголовок формы** – этот раздел находится в начале первой страницы формы и используется для размещения заголовка формы.

2 **Примечание формы** – этот раздел расположен в конце последней страницы формы и используется для размещения полезных советов, адресованных пользователю, а также другой полезной информации.

3 **Область данных** – это основная область, в которой располагаются записи данных.

Если разделы заголовка и примечания в форме не отображаются, необходимо выбрать команду Вид – Заголовок\примечание формы.

1.4.1 Изменение структуры формы

Для изменения размера области данных надо переместить указатель мыши в конец области данных и установить его там, где начинается раздел примечаний. Указатель изменит свой вид (примет вид двунаправленной стрелки). Перетащить разделительную полосу раздела «Примечание формы» вниз, увеличивая тем самым размер раздела «Область данных».

Для того чтобы увеличить размер раздела заголовка, необходимо установить указатель мыши между разделом заголовка и областью данных (указатель мыши примет вид двунаправленной стрелки), после чего перетащить вниз разделительную полосу «Области данных».

1.5 Элементы управления

Элементами управления в формах называются все объекты, с которыми можно выполнять какие-то действия, в том числе и текстовые поля.

Для выделения объекта достаточно щелкнуть на нем мышью – вокруг объекта появятся маркеры выделения.

Для выделения нескольких элементов управления необходимо установить указатель на одной из линеек (указатель изменит свой вид на жирную стрелку) и нажать кнопку мыши. Линия пройдет через форму. После того как кнопка мыши будет отпущена, все элементы управления, через которые прошла линия, будут выделены.

Если элементы управления, которые необходимо выделить, не располагаются рядом, можно выделить один элемент, а затем выделять другие, удерживая клавишу SHIFT.

Удаление выделенных элементов управления производится клавишей Delete.

Элементы управления могут быть **связанными**, **свободными** или **вычисляемыми**.

Связанный элемент управления присоединен к полю базовой таблицы

или запроса. Такие элементы управления используются для отображения, ввода или обновления значений из полей базы данных.

Для **вычисляемого элемента** управления в качестве источника данных используется выражение. В выражении могут быть использованы данные из поля базовой таблицы или запроса для формы или отчета, а также данные другого элемента управления формы или отчета. Для свободного элемента управления источника данных не существует.

Свободные элементы управления используются для вывода на экран данных, линий, прямоугольников и рисунков.

Элементы управления, которыми может пользоваться разработчик, представлены на Панели элементов. Ее открывают щелчком на соответствующей кнопке панели инструментов Microsoft Access или командой **Вид > Панель элементов** (рисунок 1.7).

Надписи предназначены для отображения в форме или отчете описательных текстов, таких как заголовки, подписи или краткие инструкции. В надписях не выводятся значения полей или выражений; они всегда являются свободными и не меняются при переходе от записи к записи.

Поля используются в форме или отчете для отображения данных из таблицы, запроса или инструкции SQL. Поле такого типа называют присоединенным, потому что оно связано с данными в поле. Кроме того, существуют свободные поля. Например, можно создать свободное поле для отображения результатов вычислений или для приема данных, вводимых пользователем. Содержимое свободного поля нигде не сохраняется.

В форме или отчете **выключатель** может быть использован как отдельный элемент управления, в котором отображаются значения логического поля из базовой таблицы, запроса.

В форме или отчете **переключатель** может быть использован как элемент управления, в котором отображаются значения логического поля из базовой таблицы, запроса.

В форме или отчете **флажок** может быть использован как отдельный элемент управления, в котором отображаются значения логического поля из базовой таблицы, запроса.

Во многих случаях удобнее выбрать значение из списка, чем вводить конкретное значение с клавиатуры по памяти. **Поле со списком** позволяет выбрать любой из этих способов ввода значения, не требуя при этом значительного места в форме. Поле со списком является комбинацией двух элементов: поля и раскрывающегося списка.

Во многих случаях удобнее выбрать значение из **списка**, чем вводить конкретное значение по памяти. Кроме того, выбор из списка позволяет быть уверенным, что введенное значение является допустимым.

Кнопки используются в формах для выполнения определенного действия или ряда действий. Например, можно создать в форме кнопку, открывающую другую форму.

Подчиненная форма – это форма, находящаяся внутри другой формы.

Первичная форма называется главной формой, а форма внутри формы называется подчиненной формой.

Подчиненным отчетом называют отчет, вставленный в другой отчет. При комбинировании отчетов один из отчетов является главным.



Рисунок 1.7 – Панель элементов

Набор вкладок используется для создания многостраничной формы.

Выбор элемента управления выполняется одним щелчком на его значке в Панели элементов, после чего следующим щелчком в поле формы отмечается место, куда он должен быть поставлен. Вместе с элементом в поле формы вставляется его присоединенная надпись. По умолчанию эта надпись стандартная, например для переключателей это Переключатель1, Переключатель2 и т.д.

чатель2 и т. д. Редактированием свойства элемента управления (доступ к свойствам открывается через контекстное меню) можно дать элементу управления более содержательную подпись.

Для создания флажков, переключателей и выключателей выполнить следующие действия: на панели элементов выбрать необходимый элемент управления; в списке полей выбрать соответствующее поле и, не отпуская мышью, перетащить его в область данных. Для выключателя надо создать надпись.

1.5.1 Перемещение элементов управления

Для перемещения элементов управления необходимо:

1 Выделить элементы управления. Передвинуть указатель мыши к краю выделенного элемента, стараясь не попасть на маркеры изменения размера. Указатель должен принять вид открытой ладони.

2 Нажать левую клавишу мыши и перетащить элемент управления в новое положение.

3 При попытке вывести элемент управления за пределы текущей области формы область формы увеличивается.

Если установить указатель мыши на левом верхнем маркере выделения элемента управления или связанной с ним надписи, то указатель примет вид указательного пальца. В таком случае перетаскивание элемента управления приводит к отдельному перемещению элемента управления и связанной с ним надписи.

1.5.2 Изменение размеров элементов управления

Изменение размеров элементов управления выполняется так же, как это принято для графических объектов – перетаскиванием маркеров изменения размера, которые расположены по углам и сторонам выделенных элементов управления.

1.5.3 Создание заголовка формы

Для создания заголовков и других свободных надписей в форме служит элемент управления Надпись (Aa).

Для создания заголовка необходимо:

1. Щелкнуть на панели элементов по кнопке **Надпись**.
2. Передвинуть указатель мыши в область заголовка, он примет вид прописной буквы «А» с перекрестием.
3. Щелкнуть на том месте, где необходимо начать ввод текста.

4. Когда текст введен, щелкнуть на любом месте формы, чтобы закрыть элемент управления.

5. Выделить созданный элемент и использовать панель форматирования для форматирования надписи.

1.5.4 Задание стиля оформления элементов управления

Для задания стиля оформления элементов управления необходимо:

- 1 Выделить элементы управления.
- 2 Для оформления границ выделенных элементов используются кнопки панели форматирования: Цвет линии\границы, Толщина линии\границы, Оформление.

1.5.5 Изменение расположения элементов управления

Фон формы образуется сеткой горизонтальных и вертикальных линий и точек. При размещении элемента управления в форме он автоматически выравнивается относительно линий сетки как по горизонтали, так и по вертикали.

Чтобы выровнять несколько элементов управления, надо их выделить и в пункте меню **Формат** выбрать команду **Выровнять**, после чего в раскрывающемся списке выбрать тип выравнивания.

1.5.6 Добавления элемента управления в форму

Если в таблицу добавлено новое поле, то в существующей форме, основанной на данной таблице, это поле будет отсутствовать. Чтобы добавить в форму новое поле, необходимо:

- выбрать команду **Вид-Список полей**;
- перетащить нужное поле из списка полей в форму и поместить его в выбранное место;
- отформатировать созданный элемент управления.

1.5.7 Создание элементов управления

С помощью кнопок панели элементов можно создавать различные элементы управления: новые поля, переключатели, флажки, поля со списком и т.д.

Для создания элемента управления **поле со списком** необходимо:

- 1 Открыть форму в режиме конструктора.
- 2 Щелкнуть по кнопке **поле со списком** на панели элементов.

3 Перетащить из списка полей в область данных поле, которое будет полем со списком. Откроется окно мастера «Создание полей со списком».

4 В **1-м шаге** мастера – выбрать способ, каким поле со списком будет получать значения: объект «поле со списком» будет использовать значения из таблицы или запроса; будет введен фиксированный набор данных; поиск записи в форме на основе значения, которое содержит поле со списком; Нажать кнопку *Далее*.

5 Во **2-м шаге** мастера – если выбран способ «а», то необходимо выбрать таблицу или запрос со значениями, которые будет содержать поле со списком. Нажать кнопку *Далее*.

6 В **3-м шаге** мастера – выбрать поле, которое содержит значения, которые следует включить в поле со списком. Отобранные поля (или поле) станут столбцами в объекте «поле со списком». Нажать кнопку *«Далее»*.

7 В **4-м шаге** мастера задать ширину столбца. Нажать кнопку *«Далее»*.

8 В **5-м шаге** мастера установить опцию «Сохранить в поле» и выбрать в раскрывающемся списке имя поля. Нажать кнопку *«Далее»*.

9 В **6-м шаге** мастера задать подпись, которую будет содержать поле со списком. Нажать кнопку *«Готово»*. При закрытии формы подтвердить изменения макета или структуры формы. Если в 1-м шаге мастера выбрать «фиксированный набор», то необходимо задать набор значений поля, которые будут использованы.

Порядок выполнения работы

1 Для таблицы **«Магазины»** создать форму, используя **автоформу в столбец** (см. п. 1.1.1);

– создать заголовок формы, поместив в область заголовка «Магазины» (см.п.1.5.3);

– заголовок отформатировать.

– сохранить форму с именами «Магазины».

– внести в базу данных несколько записей, используя созданную форму.

2 Для таблицы **«Сотрудники»** создать форму в режиме **Конструктора**:

– из таблицы «Сотрудники» перетащить поля фамилия, имя, отчество в область данных;

– создать элемент управления *поле со списком* для поля *«№ магазина»*;

– создать заголовок формы, поместив в область заголовка «Сотрудники» (см.п.1.5.3);

– заголовок отформатировать;

– сохранить форму с именем «Сотрудники»;

– внести в базу данных несколько записей, используя созданную форму.

3 Для таблиц **Товары** и **Поступление** (связь один-ко-многим) создать **многотабличную** форму, в которой таблица **Товары** – главная, таблица

Поступление – подчиненная. Для этого необходимо выполнить следующие действия:

- открыть вкладку **Формы** окна базы данных и нажать кнопку **Создать**;
 - запустить мастер форм;
 - в списке **Таблицы\запросы** выбрать таблицу **Поступление** и скопировать поля №магазина, дата, количество в список **Выбранные поля**;
 - в списке **Таблицы\запросы** выбрать таблицу **Товары** и скопировать поля наименование товара и цена в список **Выбранные поля**; Нажать кнопку **Далее**;
 - выбрать **вид** представления данных: таблицу **Товары**;
 - выбрать опцию **Подчиненные формы**. Нажать кнопку **Далее**;
 - выбрать **внешний вид** подчиненной формы. Нажать кнопку **Далее**;
 - выбрать **стиль оформления** формы. Нажать кнопку **Далее**;
 - **заголовки главной и подчиненной** форм оставить предложенные программой. В результате работы мастера будет создано две формы: главная и подчиненная;
 - установить одну из опций: открыть форму для просмотра или ввода данных или изменить макет формы. При выборе второй опции и нажатии кнопки **Готово** на экране появится проект формы;
 - создать в режиме Конструктора заголовок для главной формы, поместив в область заголовка «Товары» (см.п.1.5.3). Заголовок отформатировать.
- 4 Повторить п.3, вместо опции **Подчинённые формы** выбрать **Связанные формы**. Сравнить формы, созданные в п.3 и в п.4. В главной форме будет содержаться кнопка, щелчком на которой открывается подчиненная форма.

КОНТРОЛЬНЫЕ ВОПРОСЫ

- 1 Понятие формы.
- 2 Создание автоформы. Типы автоформ.
- 3 Создание формы с помощью мастера форм.
- 4 Создание формы в режиме Конструктора.
- 5 Создание многотабличной формы (связь один-ко-многим) с помощью мастера форм.
- 6 Панель элементов.
- 7 Создание заголовка формы.
- 8 Создание элемента управления «поле со списком».
- 9 Изменение структуры формы.
- 10 Перемещение элементов управления.

СПИСОК ЛИТЕРАТУРЫ

- 1 *Вейскас Дж.* Эффективная работа с Microsoft Access 2000. – СПб.: Питер, 2000. – 1040с.
- 2 Microsoft Access 2000: Справочник / Под ред. *Ю. Колесникова.* – СПб.: Питер, 1999. – 416с.

Учебное издание

ШИНКЕВИЧ Татьяна Леонидовна
ЛИТВИНОВИЧ Татьяна Николаевна
ЛЫЧ Юрий Павлович

Система управления базами данных MS ACCESS

Лабораторный практикум по курсу «ТОХОД»

Часть 1

Редактор Н.А. Дашкевич
Технический редактор В.Н. Кучерова

Подписано в печать 24.12.2003 г. Формат бумаги 60x84 1/16.

Бумага газетная. Гарнитура Таймс. Печать офсетная.

Усл. печ. л. 2,09. Уч.-изд. л. 2,24. Тираж 300 экз.

Зак. № . Изд. № 4038 .

Редакционно-издательский отдел БелГУТа, 246653, г. Гомель, ул. Кирова, 34.
Лицензия ЛВ № 57 от 22.10.2002 г.

Типография БелГУТа, 246022, г. Гомель, ул. Кирова, 34.
Лицензия ЛП № 360 от 25.07.1999 г.