

Документ подписан простой электронной подписью
Информация о подписи:
ФИО: Степанов Павел Иванович
Должность: Руководитель НТИ НИЯУ МИФИ
Дата подписания: 11.03.2016 14:00:00
Уникальный программный идентификатор:
8c65c591e26b2d8e460927740cf752622aa3b295

**Федеральное государственное автономное образовательное учреждение
высшего профессионального образования
Национальный исследовательский ядерный университет "МИФИ"
НОВОУРАЛЬСКИЙ ТЕХНОЛОГИЧЕСКИЙ ИНСТИТУТ**

Кафедра Автоматизации управления

СИСТЕМА УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ MICROSOFT ACCESS 2010

Методическое пособие по курсам «Информатика»,
«Информационные технологии»
для студентов всех специальностей очной формы обучения

Новоуральск 2016

Автор

Орлова Ирина Викторовна

Рецензент

кандидат физико-математических наук,
доцент кафедры ВМ Фоминых Марина Анатольевна

Система управления базами данных MicrosoftAccess 2010: Методическое пособие по курсу «Информатика», «Информационные технологии» для студентов всех специальностей очной формы обучения. Новоуральск, НТИ НИЯУ МИФИ, 2016, 68 с.

Пособие представляет собой описание одной из наиболее распространенных систем управления базами данных MicrosoftACCESS 2010.

Рассмотрены способы создания баз данных, работа с таблицами, формами, запросами и отчетами. Обсуждаемые задачи в том или ином объеме встречаются каждому студенту НТИ НИЯУ МИФИ, изучающему курс «Информатика», «Информационные технологии». Пособие рассчитано на пользователей с начальным уровнем подготовки.

Содержит 67 рисунков, 12 таблиц, 14 библиографических названий.

Пособие может использоваться при самостоятельном изучении СУБД ACCESS.

Методическое пособие рассмотрено на заседании кафедры
Протокол № 4/16 от 3июня 2016 г.

Зав. кафедрой

П.А.Дюгай

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1 ЗНАКОМСТВО С ACCESS	7
1.1 Основные понятия	7
1.2 Типы полей ACCESS	9
1.3 Проектирование базы данных	10
1.4 Пример проектирования базы данных «СКЛАД»	11
2 СОЗДАНИЕ БАЗЫ ДАННЫХ	13
2.1 Создание таблиц	13
2.2 Редактирование структуры таблиц	17
2.3 Заполнение таблицы данными	17
2.4 Редактирование записей	18
2.5 Создание поля со списком для ввода значений в таблицу	19
2.6 Связывание таблиц	20
3 ЗАПРОСЫ И ФИЛЬТРЫ	24
3.1 Запросы на выборку	24
3.2 Запросы на удаление	27
3.3 Запрос на обновление	28
3.4 Создание вычисляемых полей в запросе	29
3.5 Построитель выражений	30
3.6 Групповые операции в запросе	31
3.6.1 Вычисление итоговых значений для всех записей	31
3.6.2 Вычисление итоговых значений для групп записей в запросе	32
3.7 Фильтры	33
3.7.1 Фильтр по выделенному	33
3.7.2 Обычный фильтр	33
3.7.3 Расширенный фильтр	34
4 РАБОТА С ФОРМАМИ	35
4.1 Мастер форм	35
4.2 Работа с конструктором форм	39
4.2.1 Перенос полей и изменение их размеров	41
4.2.2 Вставка вычисляемого поля	41
4.2.3 Добавление надписи	43
4.2.4 Создание флажков, списков, полей со списками	43
4.2.6 Проведение линий и рамок	45
4.3 Создание составной формы с перечнем	46
5 ОТЧЕТЫ	49
5.1 Мастер отчетов	49
5.2 Работа с отчетом в режиме конструктора	53
5.3 Вычисления в отчетах	54
6 ЗАДАНИЯ ДЛЯ ВЫПОЛНЕНИЯ НА ЛАБОРАТОРНЫХ РАБОТАХ	57
6.1 Создание баз данных	57
6.2 Описание базы данных NATALY	58
6.3 Запросы и фильтры	59
6.4 Формы	61
6.5 Отчеты	64
7 КОНТРОЛЬНЫЕ ВОПРОСЫ ДЛЯ ОЦЕНКИ УСВОЕНИЯ МАТЕРИАЛА	65
ЛИТЕРАТУРА	67

ВВЕДЕНИЕ

Предметом изучения в данной работе является система управления базами данных (СУБД) ACCESS, которая входит в состав широко распространенного семейства офисных приложений Microsoft Office. Microsoft ACCESS на сегодняшний день является одним из самых распространенных приложений для работы с базами данных. Это связано с тем, что ACCESS обладает очень широким диапазоном средств для ввода, анализа и представления данных. Эти средства являются не только простыми и удобными, но и высокопродуктивными, что обеспечивает высокую скорость разработки приложений. Изначально ACCESS имела ряд уникальных возможностей, таких как умение сводить воедино информацию из самых разных источников (электронных таблиц, текстовых файлов, других баз данных), представление данных в удобном для пользователя виде с помощью таблиц, диаграмм, отчетов, интеграция с другими компонентами Microsoft Office. Совершенствуясь от версии к версии, ACCESS стала инструментом, который может удовлетворить потребности самых разных категорий пользователей: от новичка, которому нравится дружелюбный интерфейс системы, позволяющий ему справиться с задачами, до профессионального разработчика, который имеет весь необходимый инструментарий для построения уникального решения для конкретного предприятия среднего бизнеса. Приложение постоянно совершенствуется, практически ежегодно выпускается его новая версия и интерес к нему вспыхивает вновь. К достоинствам ACCESS можно отнести интеграцию с другими средствами Microsoft Office, возможность работы в Internet и корпоративных сетях.

Создание настоящего пособия обусловлено тем, что на компьютерах НТИ НИЯУ МИФИ в настоящее время установлена одна из последних версия пакета ACCESS 2010, которая имеет достаточно серьезные отличия от рассмотренных в работе [7] версии.

За основу пособия взята работа [7], в которую внесены изменения, связанные с новым интерфейсом пакета, добавившимися новыми возможностями. Переделаны практически все рисунки документа, добавлено несколько новых разделов. Кроме этого, при написании пособия был учтен многолетний опыт преподавания данного курса.

Области применения ACCESS обозначены достаточно ясно. Во-первых, пользователями этой системы являются непрограммирующие профессионалы – люди близкие к вычислительной технике, но не имеющие достаточного времени на ее изучение, поскольку она лежит вне области их профессиональных интересов и служит лишь подспорьем в их работе. Таких пользователей привлекает легкость изучения программы, возможность решать большинство проблем без программирования, а также средства быстрого создания приложений.

Пользователи ACCESS могут получить информацию об этой программе разными способами, например, прочтя толстое руководство или просмотрев встроенный справочник. Данное пособие предназначено для пользователя-студента, желающего как можно быстрее разобраться в ее особенностях в объеме, соответствующем учебным программам курсов «Информатика» и «Информационные технологии» НТИ НИЯУ МИФИ.

Пособие построено в соответствии с идеей, что даже простой пример выполнения некоторой операции может дать больше, чем целая теоретическая глава руководства пользователя. Второй принцип данной работы – объяснить, где что лежит. Если вы знаете, где программа содержит инструмент для выполнения нужной операции, то вы разберетесь, как им пользоваться. Но если вы не знаете, что, вообще, умеет делать ACCESS, то вам непонятно главное – что нужно изучать и о чем спрашивать.

В силу вышесказанного, пособие ни в коей мере не претендует на полноту изложения. Остались нерассмотренными такие темы, как макросы, модули, взаимодействие с дру-

гими приложениями Microsoft Office. Основную цель автор видел в создании достаточно компактного описания, позволяющего студенту освоить основы работы в ACCESS.

Рисунки в пособии подобраны так, чтобы они были не только иллюстрацией к тексту, но и служили дополнительным источником полезной информации. Например, часто приводятся изображения вкладок и контекстных меню.

Пособие можно просто читать, но автор рекомендует работать с ACCESS за компьютером. База данных «СКЛАД», описанная в работе, хранится в каталоге WORK на устройстве D: или G: всех ПЭВМ отдела информатизации.

В последнем разделе работы приведены задания для студентов, выполняющих лабораторные работы по разделу ACCESS курсов «Информатика» и «Информационные технологии». Задания имеют разную степень сложности и объема, что обусловлено различным временем, отводимым на изучение ACCESS студентами различных специальностей.

При описании работы с любыми программами требуется некоторое средство описания последовательности действий, которые пользователь должен выполнять на компьютере. Принятая в настоящем пособии система таких действий приведена в таблице 1.

Таблица 1 - Система обозначений действий пользователя

Используемое обозначение	Выполняемое действие
➤ Название команды	Выбор в текущем меню команды с указанным названием, то есть либо позиционирование подсветки на указанной команде с помощью клавиш управления курсором и нажатие клавиши <Enter>, либо позиционирование указателя мыши на указанной команде и однократное нажатие левой кнопки мыши.
☐ Название кнопки	Нажатие кнопки с указанным названием в активном диалоговом окне, то есть либо позиционирование выделяющей рамки на указанной кнопке с помощью клавиши <Tab> и нажатие клавиши <Enter>, либо позиционирование указателя мыши на указанной кнопке и однократное нажатие левой кнопки мыши.
⇩ Название раскрывающегося списка или палитры	Развертывание раскрывающегося списка или палитры, то есть позиционирование указателя мыши на кнопке раскрытия списка (кнопка со стрелкой, направленной вниз) и однократное нажатие левой кнопки мыши.
☞ Фрагмент текста, ячейка или диапазон ячеек, элемент списка или раскрывающегося списка	Выделение указанного фрагмента текста, диапазона ячеек или элемента списка, то есть его выделение либо с помощью клавиш управления курсором при нажатой и удерживаемой клавише <Shift>, либо при помощи мыши при нажатой и удерживаемой левой кнопке мыши.
☐ Название вкладки	Раскрытие вкладки с указанным названием в активном диалоговом окне, то есть позиционирование указателя мыши на указанном ярлычке и однократное нажатие левой кнопки мыши.
Название поля := <u>значение</u>	Ввод значения с клавиатуры в текстовое поле ввода, раскрывающийся список или счетчик. Значение счетчика может также изменяться с помощью кнопок регуляторов. Вводимые с клавиатуры значения выделены в наших обозначениях подчеркиванием.

Используемое обозначение	Выполняемое действие
☉ Название переключателя	Выбор переключателя с указанным названием в активном диалоговом окне, то есть позиционирование указателя мыши на указанном переключателе и однократное нажатие левой кнопки мыши.
☒ Название флажка	Отметка флажка с указанным названием в активном диалоговом окне, то есть позиционирование указателя мыши на указанном флажке и однократное нажатие левой кнопки мыши.
<Клавиша>	Нажатие соответствующей клавиши на клавиатуре.
<Клавиша1>/<Клавиша2>	Нажатие Клавиши1 и нажатие Клавиши2 при удерживаемой Клавише1.

По большей части выполняемые действия далее написаны в терминах команд меню — так гораздо проще *описывать* последовательность действий пользователя — однако *выполнять* эти действия зачастую гораздо проще с помощью эквивалентных кнопок на панелях инструментов.

Кроме специальных обозначений, показанных в таблице 1, в работе используются текстовые выделения и обозначения.

Определяющие вхождения новых понятий выделены **полужирным шрифтом**; термины, т.е. названия интерфейсных элементов: окон, кнопок, панелей инструментов выделены шрифтом **Courier**.

Имена объектов баз данных (таблиц, форм и т.п.) записаны в кавычках заглавными буквами, например, «ТОВАРЫ».

Имена полей записаны в кавычках, первая буква заглавная, остальные - строчные, например, «Покупатель».

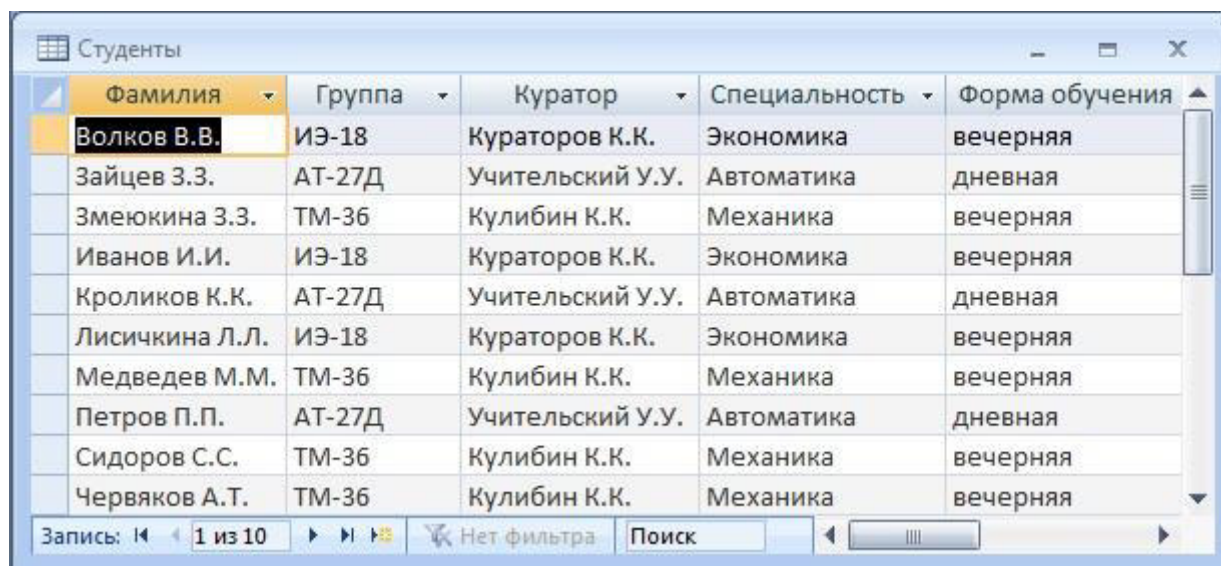
1 ЗНАКОМСТВО С ACCESS

1.1 Основные понятия

Системы управления базами данных имеют более чем сорокалетнюю историю развития с сохранением преемственности и устойчивости традиций. В основе СУБД лежит концепция **модели данных**, т.е. некоторой абстракции представления данных. В большинстве случаев предполагается, что данные представлены в виде **файлов**, состоящих из **записей**. Структура всех записей в файлах одинакова, а количество записей в файле является переменным. Элементы данных, из которых состоит каждая запись, называются **полями**. Поскольку во всех записях обязательно имеются одни и те же поля (с разными значениями), полям дают уникальные **имена**.

Многие практически важные случаи хорошо укладываются в такое представление данных. Например, в отделе кадров информация о сотрудниках имеет такую природу: сотрудников принимают и увольняют, но форма личного листка по учету кадров остается неизменной для каждого сотрудника. Товарно-материальные ценности приходят и уходят, но форма инвентарной карточки также остается неизменной. Даже из этих примеров ясно, что СУБД является подходящим средством во всех случаях, когда исходную информацию можно представить в виде таблицы постоянной структуры, но неопределенной длины, или в виде картотеки, содержащей неопределенное количество карточек постоянной структуры.

На рисунке 1.1 показан фрагмент таблицы ACCESS, которая делится на строки и столбцы, а значения вводятся в ячейки, разделенные линиями сетки. Каждый столбец представляет собой поле, а каждая строка – запись. Каждое поле таблицы может содержать значения данных только одного типа – текст, числа, даты и т.д., а каждая запись содержит информацию об одном элементе (например, сведения о сотруднике или какой-то материальной ценности).



Фамилия	Группа	Куратор	Специальность	Форма обучения
Волков В.В.	ИЭ-18	Кураторов К.К.	Экономика	вечерняя
Зайцев З.З.	АТ-27Д	Учительский У.У.	Автоматика	дневная
Змеюкина З.З.	ТМ-36	Кулибин К.К.	Механика	вечерняя
Иванов И.И.	ИЭ-18	Кураторов К.К.	Экономика	вечерняя
Кроликов К.К.	АТ-27Д	Учительский У.У.	Автоматика	дневная
Лисичкина Л.Л.	ИЭ-18	Кураторов К.К.	Экономика	вечерняя
Медведев М.М.	ТМ-36	Кулибин К.К.	Механика	вечерняя
Петров П.П.	АТ-27Д	Учительский У.У.	Автоматика	дневная
Сидоров С.С.	ТМ-36	Кулибин К.К.	Механика	вечерняя
Червяков А.Т.	ТМ-36	Кулибин К.К.	Механика	вечерняя

Рисунок 1.1 - Фрагмент таблицы ACCESS

Рассматривают два основных вида баз данных (БД): **плоские таблицы** и **реляционные базы**.

В плоских таблицах вся взаимосвязанная информация должна находиться в одной таблице. Это означает, что любые данные, повторяющиеся в нескольких записях, должны присутствовать в каждой из этих записей. Таблица, представленная на рисунке 1.1, является именно плоской таблицей.

В реляционных базах, к которым относится ACCESS, используется несколько разных таблиц, между которыми устанавливаются **связи** (relations). Они позволяют ввести информацию в одну таблицу и связать ее с записями в другой через специальное поле. На рисунке 1.2 показана та же информация, что и на рисунке 1.1, но в последнем случае данные разделены на две таблицы, между которыми установлена связь. Дублировавшаяся информация на рисунке 1.2 встречается только один раз во второй таблице.

Поле связи

Фамилия	Группа
Иванов И.И.	2ИЗ-18
Петров П.П.	2АТ-27Д
Сидоров С.С.	2ТМ-36
Медведев М.М.	2ТМ-36
Зайцев З.З.	2АТ-27Д
Волков В.В.	2ИЗ-18
Кроликов К.К.	2АТ-27Д
Лисицына Л.Л.	2ИЗ-18
Змеюкина З.З.	2ТМ-36
Червяков Ч.Ч.	2ТМ-36

Группа	Куратор	Специальность	Форма обучения
2ИЗ-18	Кураторов К.К.	Экономика	вечерняя
2АТ-27Д	Учительский У.	Автоматика	дневная
2ТМ-36	Кулибин К.К.	Механика	вечерняя

Рисунок 1.2 - Пример реляционной структуры БД

Хранение данных в связанных таблицах обладает рядом преимуществ:

- ◆ экономия времени, т.к. одни и те же данные не приходится вводить несколько раз;
- ◆ уменьшение размера базы данных;
- ◆ существенное сокращение количества ошибок из-за уменьшения объема вводимой информации.

Все СУБД поддерживают в той или иной форме четыре основные операции:

- ◆ **добавить** в БД одну или несколько записей;
- ◆ **удалить** из БД одну или несколько записей;
- ◆ **найти** в БД записи, удовлетворяющие заданному условию;
- ◆ **обновить** в БД значения некоторых полей в одной или нескольких записях.

Все объекты, относящиеся к конкретной БД, ACCESS хранит в одном файле. Основными объектами ACCESS являются:

- ◆ **таблицы**, в которых хранится информация;
- ◆ **запросы**, с помощью которых вы запрашиваете разные сведения;
- ◆ **формы и отчеты**, используемые для представления данных;
- ◆ **страницы, модули и макросы** для автоматизации работы с базой.

Все объекты в базе данных упорядочиваются по своему типу и отображаются в левой части окна Microsoft ACCESS, как показано на рисунке 1.3. В правой части окна ACCESS отображается содержимое открытого объекта (таблицы, формы, отчета и т.д.).

Работа с объектами в БД ACCESS происходит в двух режимах: конструктора и данных (таблицы, формы, просмотра).

Перед тем, как объект БД сможет работать с вашей информацией, его необходимо создать. В отличие от открытия пустых файлов в Word или пустых листов в Excel, все объекты ACCESS предварительно должны быть сконструированы. В это понятие входит процесс создания объекта и задание его свойств. Именно конструкция объекта определяет, что вы увидите при работе с данными, используя этот объект.

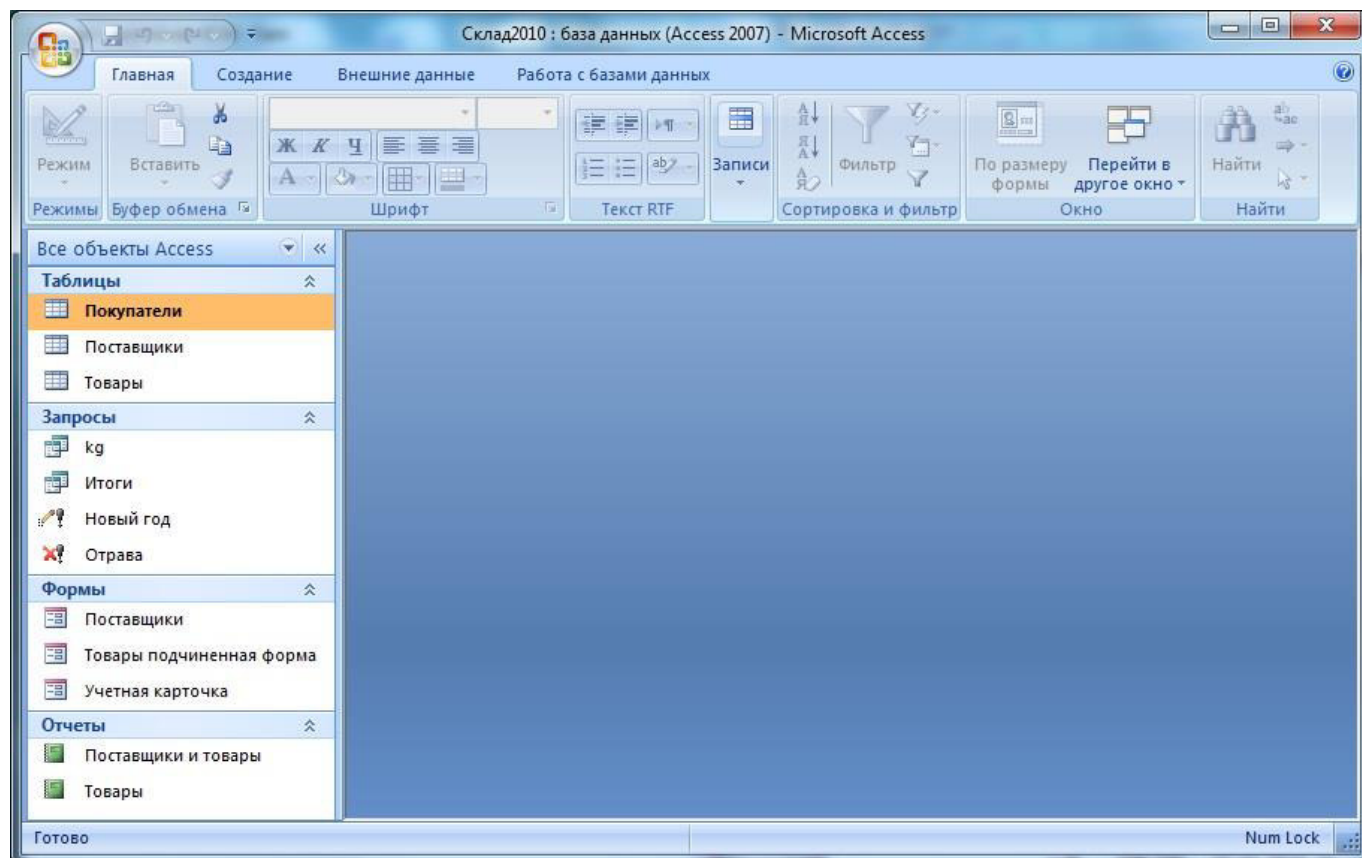


Рисунок 1.3 - Окно СУБД ACCESS 2007

После этапа конструирования наступает рабочая фаза, во время которой объекты БД используются для ввода и редактирования данных, извлечения информации из БД при помощи запросов и отчетов.

Одна из очень полезных черт ACCESS заключается в том, что при работе с объектами различных типов, а также при переходе от режима конструктора к рабочему, вид главного меню и панели инструментов изменяется в соответствии с теми задачами, которые приходится выполнять.



ВНИМАНИЕ

При работе с ACCESS следует помнить о том, что любые изменения, вносимые в содержимое таблиц (т.е. изменение данных) сохраняются автоматически, как только такие изменения сделаны, специального сохранения файла при этом не требуется. С другой стороны, макеты различных объектов должны сохраняться отдельной командой.

1.2 Типы полей ACCESS

В каждом поле БД могут храниться данные одного типа, поэтому при создании своей базы следует заранее определить тип соответствующего поля. Реально ACCESS предоставляет в ваше распоряжение следующие восемь типов полей.

Текстовый. Это поле предназначено для хранения букв, цифр или любых их комбинаций длиной до 255 символов. Количество хранимых символов зависит от значения свойства **Размер поля**, о котором будет рассказано ниже.

Поле МЕМО. Поля этого типа подобны текстовым, но могут содержать до 65535 символов. Для этого типа не указывается размер, он равен тому количеству символов, которое туда вводится.

Числовой. Содержит числовые значения, которые могут использоваться в математических выражениях.

Размер числового поля может быть выбран из следующего списка:

- ◆ Байт – целые числа от 0 до 255;
- ◆ Целое – целые числа от – 32768 до 32767;
- ◆ Длинное целое – целые числа от –2 147 483 648 до 2 147 483 647;
- ◆ Одинарное с плавающей точкой (4 байта) – числа в пределах от $\approx -3.4 \cdot 10^{38}$ до $\approx 3.4 \cdot 10^{38}$;
- ◆ Двойное с плавающей точкой (8 байт) – числа в пределах от $\approx -1.8 \cdot 10^{308}$ до $\approx 1.8 \cdot 10^{308}$.

Дата/Время. Поле используется для хранения информации о дате или/и времени какого-либо события. Формат поля можно установить из списка возможных значений.

Денежный. По умолчанию этот формат представляет собой числа, записанные с двумя знаками после запятой, с разделением групп разрядов и обозначением «р.» в конце.

Счетчик. Поле, значение которого автоматически увеличивается при каждом добавлении новой записи. ACCESS не использует повторно номера, которые освобождаются при удалении записей. Кроме этого, вы не можете сами изменить значение этого поля.

Логический. Наиболее эффективный способ хранения единственного значения, соответствующего выбору «Да/Нет», «Истина/Ложь» и т.д. Значение чаще всего представляется в виде флажка.

Поле объекта OLE. Содержит объект OLE (лист Excel, документ Word, рисунок, звук и т.д.), который может быть вставлен командой «Вставка».

1.3 Проектирование базы данных

Первым и главным вопросом, подлежащим решению при создании БД, является **проектирование**, т.е. определение того, какие таблицы будут входить в базу, какие в них будут находиться поля, как будут связаны таблицы, какая информация будет храниться в таблицах базы, а какая будет вычисляться с помощью запросов и т.д. Этот вопрос является главным, а все остальные вопросы решаются почти автоматически при наличии легко приобретаемого навыка.

Данная часть работы выполняется не за компьютером, а перед запуском ACCESS и состоит из нескольких последовательных этапов.

Определение содержимого полей. На первом этапе следует определить, какие поля с информацией должны быть включены в БД. Необходимо постараться указать все элементы, которые могут понадобиться в работе, но с другой стороны необходимо соблюдать баланс между полезностью информации и простотой ее обработки.

Выбор типа информации в полях. Далее необходимо определить тип информации, хранящейся в каждом поле, и характеристики соответствующего типа.

Определение структуры БД. Здесь должен быть решен вопрос о том, как должны быть организованы поля в БД. Необходимо определить, каким образом поля будут распределены по таблицам и в каком порядке.

Назначение первичных ключей и индексов. Первичным ключом называется одно или несколько полей, которые однозначно определяют каждую запись в таблице. Наличие индекса помогает ACCESS быстрее находить и сортировать записи. Поля, используемые в качестве первичного ключа, индексируются автоматически, но можно составить отдельный индекс и для других полей, которые предполагается использовать для поиска и сортировки. Первичные ключи и индексы обычно указываются в таблице во время ее создания.

Определение порядка ввода данных. На этом этапе следует продумать, какие таблицы должны быть заполнены раньше остальных, чтобы использовать их значения в качестве списка для другой таблицы. Иногда следует составить и ввести список значений какого-либо поля, чтобы использовать эти значения при вводе. Достаточно часто существуют стандартные значения каких-либо полей, которые ACCESS должен ввести в какое-либо поле автоматически (значение по умолчанию).

Следует отметить, что если в некоторых полях должны содержаться результаты выполнения математических операций над другими полями таблиц, их не следует включать в структуру таблицы. Вычисляемые поля могут входить в запросы, формы, отчеты и при этом не быть частью таблицы. Такое решение снимает необходимость в хранении ненужной информации.

1.4 Пример проектирования базы данных «СКЛАД»

Пусть имеется склад, на котором хранятся товары. Товары имеют определенное наименование и цену. Товары поступают на склад и уходят со склада. Проектируемая база данных должна позволять получать информацию о текущем состоянии склада, т.е. сведения о количестве и стоимости товаров на складе. В такой общей постановке задача перекрывает едва ли не половину реально используемых приложений СУБД. В качестве упрощения не будем учитывать «пересортицу», т.е. тот факт, что в реальности разные товары могут иметь одно наименование, и в то же время одни и те же товары могут иметь разные цены.

Кроме рассмотренной информации в проектируемой базе данных должны храниться сведения о внешних связях, например, о поставщиках и потребителях товаров.

В такой постановке задачи всю информацию удобно хранить в трех отдельных таблицах: сведения о товарах, информация о поставщиках и данные о потребителях.

Характеристики полей этих таблиц представлены в таблицах 1.1 – 1.3.

При вводе данных, очевидно, следует сначала заполнить таблицы «ПОКУПАТЕЛИ» и «ПОСТАВЩИКИ» для того, чтобы значения соответствующих полей в таблице «ТОВАРЫ» («Клиент» и «Поставщик») можно было взять уже из готовых таблиц.

Таблица 1.1 - Характеристики полей таблицы «ТОВАРЫ»

Имя поля	Тип данных	Длина поля	Примечание
Номер	Счетчик		
Товар	Текстовый	20	Наименование товара (Ключевое поле)
Ед_изм	Текстовый	3	Единица измерения
Кол-во	Числовой	целое	Количество товара
Цена	Денежный		Цена единицы товара
Поставщик	Текстовый	8	Краткое имя поставщика товара (поле связи с таблицей «ПОСТАВЩИКИ»)
Клиент	Текстовый	6	Краткое имя покупателя товара (поле связи с таблицей «ПОКУПАТЕЛИ»)
Годен до	Дата/Время		Срок годности товара
Сертификат	Логический		Наличие сертификата
Описание	МЕМО		Описание товара

Таблица 1.2 - Характеристика полей таблицы «ПОКУПАТЕЛИ»

Имя поля	Тип данных	Длина поля	Примечание
Клиент	Текстовый	6	Краткое имя покупателя товара (Ключевое поле)
Название	Текстовый	24	Полное наименование покупателя
Обращаться к	Текстовый	15	Лицо из фирмы «Покупатель», с которым осуществляется связь
Должность	Текстовый	12	Должность соответствующего лица
Адрес	Текстовый	20	Адрес покупателя

Таблица 1.3 - Характеристика полей таблицы «ПОСТАВЩИКИ»

Имя поля	Тип данных	Длина поля	Примечание
Поставщик	Текстовый	8	Краткое имя поставщика товара (Ключевое поле)
Название	Текстовый	24	Полное наименование поставщика
Телефон	Текстовый	9	Телефон поставщика
Адрес	Текстовый	30	Адрес поставщика

2 СОЗДАНИЕ БАЗЫ ДАННЫХ

2.1 Создание таблиц

Начнем создание нашей информационно-справочной системы. При запуске ACCESS открывается диалоговое окно для определения режима работы (рисунок 2.1).

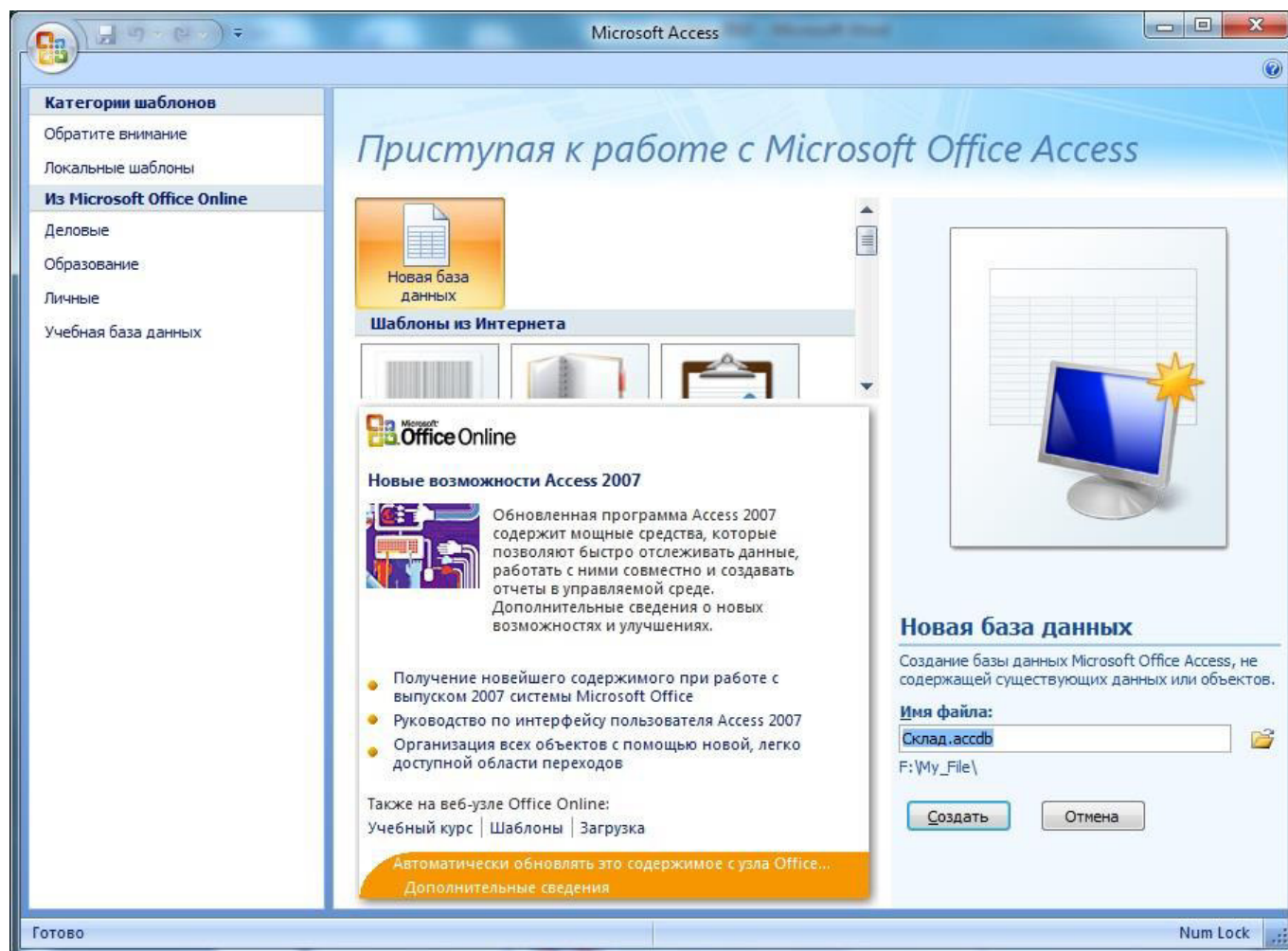


Рисунок 2.1 - Диалоговое окно, открывающееся при запуске ACCESS

Сначала создадим новую базу данных:

☐ Создать файл ☒ Новая база данных Имя файла: Склад ☐ Создать.

После этого открывается окно **Склад: База данных(Access 2007) – Microsoft Access** (рисунок 2.2) с **Таблицей1**, открытой в режиме таблицы.

Таблицу можно создавать в режиме таблицы, добавляя поля и заполняя их данными, но удобнее перейти в режим конструктора, сохранив таблицу с нужным именем (в нашем случае – **ТОВАРЫ**).

В режиме конструктора на экране появляется пустой бланк характеристик полей БД, который мы должны заполнить.

Каждое поле обладает индивидуальными свойствами, задавая которые можно установить, как должны сохраняться, отображаться и обрабатываться данные. Набор свойств поля зависит от выбранного типа данных.

В таблице 2.1 приведено описание основных свойств полей.

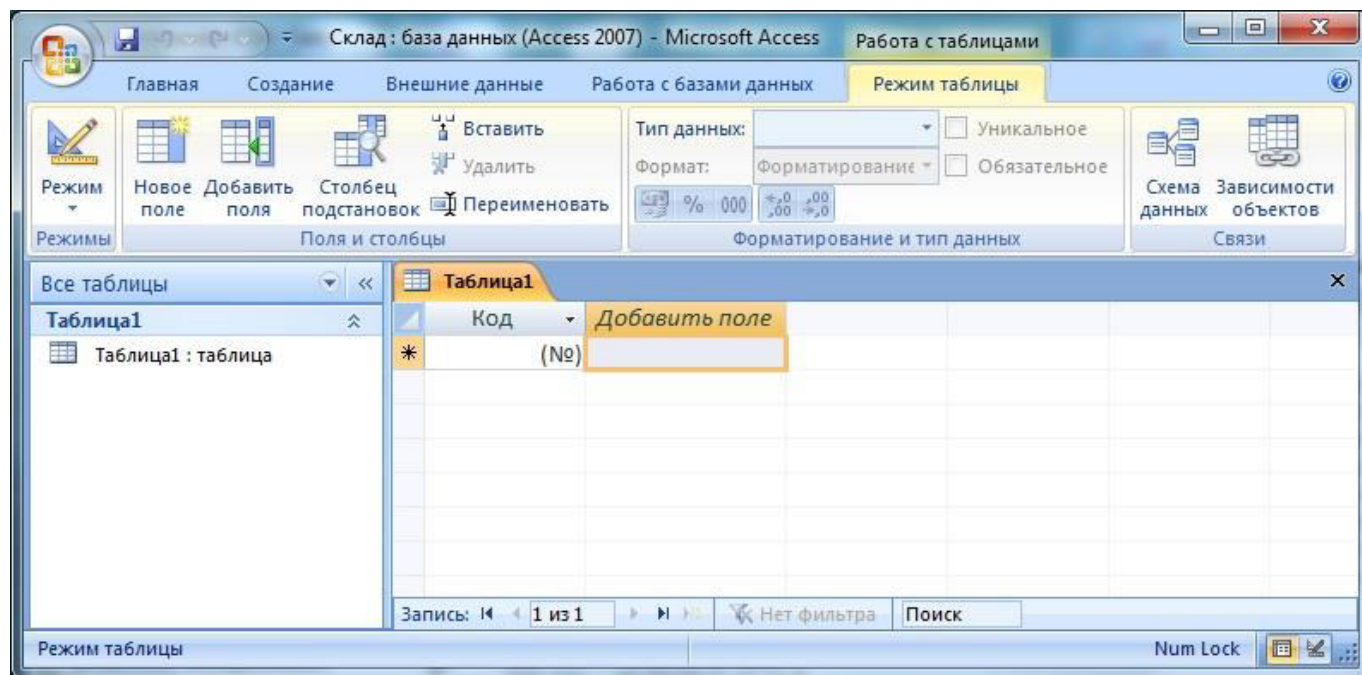


Рисунок 2.2 - Диалоговое окно, открывающееся при запуске ACCESS

Таблица 2.1 - Основные свойства полей базы данных

Свойство	Описание
Размер поля	Определяет максимальную длину текстового или числового поля. При создании слишком большого размера неэкономно расходуется память, а при слишком маленьком размере может исказиться содержимое поля.
Формат поля	Устанавливает формат отображения данных в форме или запросе.
Число десятичных знаков	Определяет количество десятичных разрядов в дробной части десятичного числа.
Подпись	Необязательная характеристика, содержащая надпись, которая выводится рядом с полем в форме или отчете. Эта надпись чаще всего содержит поясняющие сведения о поле.
Значение по умолчанию	Содержит значение, устанавливаемое по умолчанию в соответствующем поле таблицы.
Условие на значение	Определяет множество значений, которые пользователь может вводить в это поле. При вводе иного значения поле считается незаполненным и выдается сообщение об ошибке.
Сообщение об ошибке	Определяет сообщение, которое появляется на экране, если введено значение, не входящее во множество допустимых значений.
Обязательное поле	Установка, указывающая на то, что данное поле обязательно следует заполнить.
Пустые строки	Установка, определяющая допуск наличия в данном поле пустых (незаполненных) строк.
Индексированное поле	Определяет необходимость создания индексов для ускорения поиска по данному полю.

Смысл всех этих свойств понятен (тем более что по каждой графе при ее заполнении выводится краткая справка), а назначение все они имеют одно и то же: предоставить ACCESS дополнительную информацию для лучшей работы с вашими данными, поэтому принцип заполнения очень прост: если вы знаете (или думаете, что знаете), как заполнить графу, то – заполняете, если нет – то оставьте незаполненной, ACCESS что-нибудь придумает сам.

После определения всех полей таблицы следует указать, по меньшей мере, одно поле для использования его в качестве первичного ключа (ключевое поле), что не позволит вводить в таблицу повторяющиеся записи, поскольку поле первичного ключа должно содержать однозначный идентификатор для каждой записи. Это поле не может содержать одинаковое значение для двух различных записей.

В разработанной таблице «ТОВАРЫ» с учетом упрощений, сделанных в разделе 1.4, первичным ключом выберем поле **Товар**. Ключ задается с помощью команды

 **Конструктор**  **Ключевое поле** (с изображением ключа)

или с помощью соответствующей команды из контекстного меню для выбранного поля.

Достаточно часто в качестве ключевого поля в БД используется поле типа **Номер**, т.к. по его свойствам значение такого поля повторяться не может.

Структура таблицы «ТОВАРЫ» показана на рисунке 2.3.

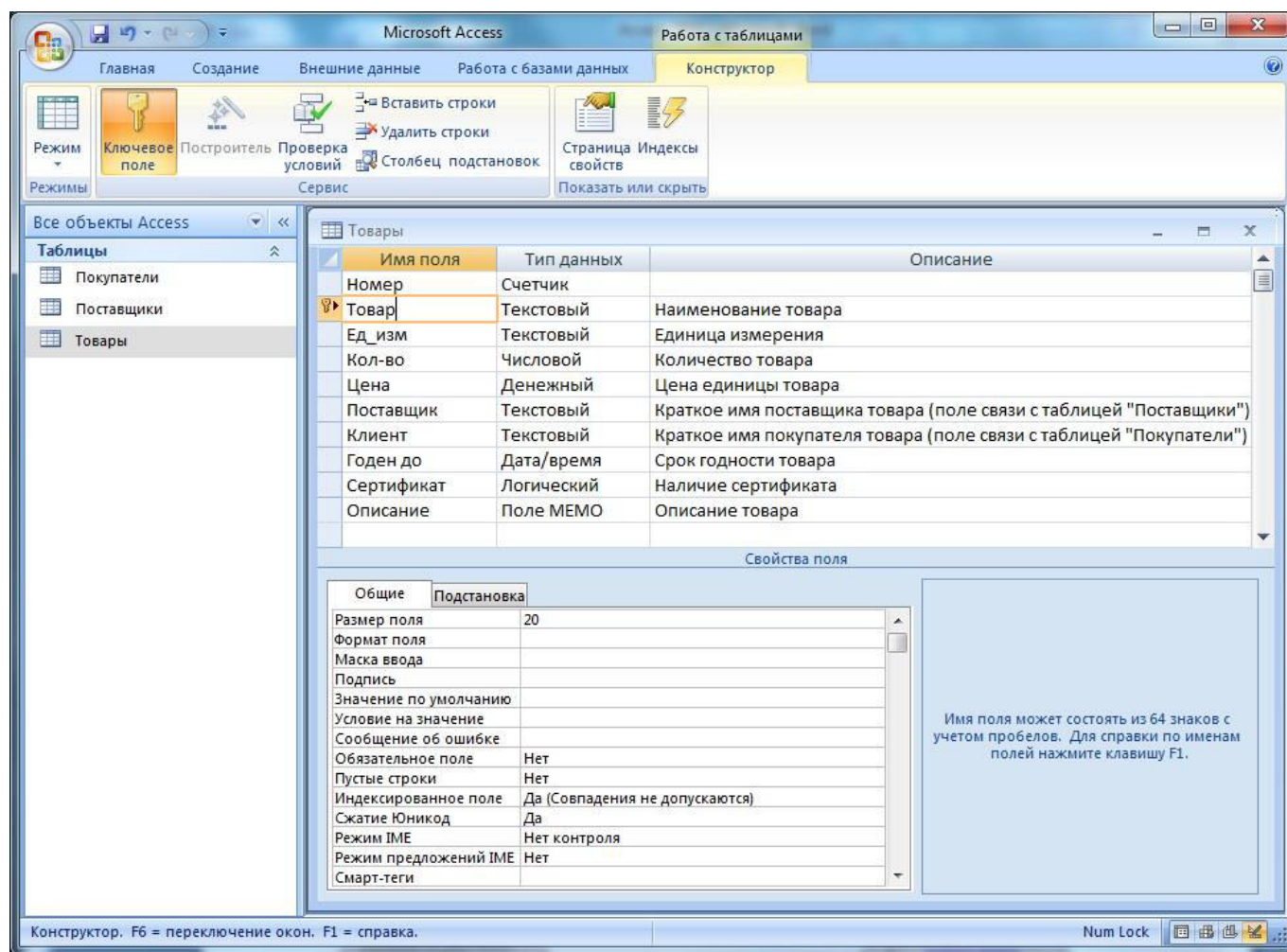


Рисунок 2.3 - Структура таблицы «ТОВАРЫ»

Созданную таблицу необходимо сохранить с помощью команды:

 **Файл**  **Сохранить**

После завершения создания таблицы «ТОВАРЫ» создаем еще две таблицы: «ПОСТАВЩИКИ» и «ПОКУПАТЕЛИ», которые будут содержать информацию о внешних связях нашей БД.

Для создания каждой из этих таблиц выполняем последовательность действий:

□ Создание □ Таблицы □ Конструктор таблиц

Структура этих таблиц показана на рисунках 2.4 и 2.5.

Имя поля	Тип данных	Описание
Поставщик	Текстовый	Краткое имя поставщика товара
Название	Текстовый	Полное наименование поставщика
Телефон	Текстовый	Телефон поставщика
Адрес	Текстовый	Адрес поставщика

Свойства поля

Общие	Подстановка
Размер поля	8
Формат поля	
Маска ввода	
Подпись	
Значение по умолчанию	
Условие на значение	
Сообщение об ошибке	
Обязательное поле	Нет
Пустые строки	Нет
Индексированное поле	Да (Совпадения не допускаются)
Сжатие Юникод	Да
Режим IME	Нет контроля
Режим предложений IME	Нет
Смарт-теги	

Имя поля может состоять из 64 знаков с учетом пробелов. Для справки по именам полей нажмите клавишу F1.

Рисунок 2.4 - Структура таблицы «ПОСТАВЩИКИ»

Имя поля	Тип данных	Описание
Клиент	Текстовый	Краткое имя покупателя товара
Название	Текстовый	Полное наименование покупателя
Обращаться к	Текстовый	Лицо из фирмы "Покупатель", с которым осуществляется связь
Должность	Текстовый	Должность соответствующего лица
Адрес	Текстовый	Адрес покупателя

Свойства поля

Общие	Подстановка
Размер поля	6
Формат поля	
Маска ввода	
Подпись	
Значение по умолчанию	
Условие на значение	
Сообщение об ошибке	
Обязательное поле	Нет
Пустые строки	Нет
Индексированное поле	Да (Совпадения не допускаются)
Сжатие Юникод	Да
Режим IME	Нет контроля
Режим предложений IME	Нет
Смарт-теги	

Имя поля может состоять из 64 знаков с учетом пробелов. Для справки по именам полей нажмите клавишу F1.

Рисунок 2.5 - Структура таблицы «ПОКУПАТЕЛИ»

2.2 Редактирование структуры таблиц

Готовую структуру таблиц можно модифицировать, например, добавлять (удалять) поля или изменять их параметры. При этом все исправления желательно внести в структуру до начала заполнения базы данных, поскольку попытка изменить параметры полей заполненной базы может повлечь за собой потерю или искажение данных.

Для вставки нового поля следует установить курсор в строке, перед которой должна быть введена новая строка, и выбрать команду

 **Конструктор**  **Сервис**  **Вставить строки**

либо соответствующую команду из контекстного меню.

Если из структуры таблицы необходимо удалить поле, то следует маркировать соответствующую строку и аналогично выбрать команду **Удалить строки**. Строка таблицы маркируется с помощью столбца маркировки. Когда указатель мыши попадает в этот столбец, он приобретает вид стрелки, указывающей вправо.

При удалении поля из заполненной таблицы удаляются все данные, находящиеся в нем. Перед удалением данных ACCESS предупреждает об этом и предоставляет возможность отказаться от удаления.

Изменение типа данных поля после ввода данных в таблицу иногда сопровождается длительной процедурой преобразования данных в момент сохранения таблицы. Если ACCESS не сможет выполнить преобразование без ошибок, часть данных будет потеряна или искажена. Увеличение размера поля не оказывает влияния на содержащиеся в нем данные, однако если размер поля уменьшить, то его содержимое может исказиться. Так, например, если размер текстового поля уменьшить с 50 символов до 20, то записи, которые содержат больше 20 символов, будут содержать первые 20 исходных символов, а остальные будут отброшены.

2.3 Заполнение таблицы данными

Для добавления записей в таблицу нужно выполнить двойной щелчок по имени нужной таблицы в окне базы данных (рисунок 1.3) или выбрать команду **Открыть** из контекстного меню. Если в данный момент открыто окно **Конструктора**, следует установить **Режим таблицы**, вызвав соответствующую команду из списка **Режим** или из контекстного меню.

В режиме таблицы переход к следующему полю осуществляется с помощью клавиши <Tab>, а к предыдущему – с помощью комбинации клавиш <Shift>/<Tab>. С помощью клавиш управления курсором <↑> и <↓> выполняется перемещение от одной строки к другой. После окончания ввода и нажатия клавиши <Tab> ACCESS автоматически сохранит только что введенные данные в файле.

Перейти к нужной записи также можно с помощью кнопок перехода (рисунок 2.6), либо задать номер нужной записи в соответствующем поле и нажать <Enter>. А также можно просто щелкнуть мышью в том поле, где должен находиться курсор.

Заполненные данными таблицы «ПОКУПАТЕЛИ» и «ПОСТАВЩИКИ» показаны на рисунках 2.6 и 2.7.

Заполнение таблицы «ТОВАРЫ» отложим до создания связей между таблицами.

Покупатели					
	Клиент	Название	Обращаться к	Должность	Адрес
+	Абзац	Кафе "Абзац"	Антон Крутой	Менеджер	Бертольда Шварца, 33
+	Амбра	Закусочная "Амброзия"	Таисия Ченчик	Директор	Деникина, 107
+	Баран	АО "Бараний рог"	Фёдор Павлович	Руководитель	Космонавтов, 77
+	Лебедь	Ресторан "Белый лебедь"	Белла Лебедева	Менеджер	Чекистов, 123
+	Пиран	Бистро "Пиранья"	Иван Сусанин	Директор	Северная, 17
*					
Запись: 1 из 5					
Нет фильтра					
Поиск					

Новая (пустая) запись
 Последняя запись
 Следующая запись
 Предыдущая запись
 Первая запись

Рисунок 2.6 - Заполненная таблица «ПОКУПАТЕЛИ»

Поставщики				
	Поставщик	Название	Телефон	Адрес
+	Delikate	"Французский деликатес"	470-33902	с. Тараньки Ивановской обл.
+	Fruct	US Fruits	838-37372	New York, Kosmonavtov street 2
+	Selderey	ТО "Пряности и приправы"	950-24251	Архангельск, Северная 44
+	Tokyo	Tokyo Traders	535-64635	Tokyo, Vakimary Tymakuka 256
+	Восток	АО "Восточная звезда"	77-31-234	Одесса, Дерибасовская 85
+	Тибет	АО "Тибетский продукт"	737-23-23	India, Punjab, Uhtilysy 88
*				
Запись: 1 из 6				
Нет фильтра				
Поиск				

Рисунок 2.7 - Заполненная таблица «ПОСТАВЩИКИ»

2.4 Редактирование записей

Пользователь может добавлять в таблицу новые записи и удалять или редактировать существующие.

Маркирование записей

Перед редактированием запись необходимо маркировать, что выполняется с помощью щелчка мышью в столбце маркировки (слева от записи). После выполнения этого действия будет отмечена активная запись, т.е. та, в которой находится курсор ввода. Для отмены маркировки следует выполнить щелчок в любом месте таблицы. Маркировать несколько записей, расположенных рядом, можно с помощью буксировки мышью в столбце маркировки, либо выполнить щелчок мышью слева от первой(или последней) из них, а затем, удерживая клавишу **<Shift>**, выполнить щелчок мышью слева от последней (или первой) записи нужного диапазона.

Перемещение и копирование записей

Чтобы переместить или скопировать маркированную запись (группу записей), её следует поместить в буфер обмена с помощью команд **Вырезать** (**Копировать**) из кон-

текстного меню или соответствующих кнопок панели инструментов, а затем **вставить** в любую позицию данной или другой таблицы.

Копирование записи сопровождается копированием всех её атрибутов. При вставке записи в другую таблицу ACCESS предполагает, что целевая таблица имеет такую же структуру, что и таблица-источник.

Удаление записей

Если из таблицы нужно удалить запись или несколько записей, нужно их маркировать и воспользоваться командой **Удалить запись** из контекстного меню, либо командой **Удалить** на вкладке **Главная** (группа **Запись**). При этом необходимо соблюдать осторожность, поскольку после удаления записей команда **Отменить** невозможна.

Закреть таблицу можно посредством команды **Закреть** из контекстного меню таблицы или щелчком по кнопке **Закреть** окна таблицы. Сохранять данные не требуется, так как программа ACCESS автоматически помещает записи в файл на жестком диске по мере их ввода.

2.5 Создание поля со списком для ввода значений в таблицу

В том случае, если значение какого-либо поля таблицы повторяется, удобнее вводить это значение путем выбора из списка.

В качестве списка можно использовать фиксированный набор значений или набор значений из другой таблицы. Рассмотрим оба способа создания списка.

Использование фиксированного набора значений.

Создадим поле для ввода значений «Ед_изм» в таблице «ТОВАРЫ». Для этого откроем таблицу «ТОВАРЫ» в режиме конструктора. В открывшемся окне конструктора для поля «Ед_изм» в колонке **Тип данных** в списке доступных типов полей выберем элемент **Мастер подстановок**.

В открывшемся при этом диалоговом окне **Создание подстановки** активизируем опцию **Будет введен фиксированный набор значений** и нажмем кнопку **Далее**. В следующем диалоговом окне **Создание подстановки** введем набор значений, которые может принимать поле «Ед_изм» (рисунок 2.8), и нажмем кнопку **Готово**.

Использование набора значений из другой таблицы.

Для создания списка ввода поля «Клиент» таблицы «ТОВАРЫ» из значений поля другой таблицы (поле «Клиент» таблицы «ПОКУПАТЕЛИ») аналогично выбираем **Тип данных** → **Мастер подстановок**, а затем в окне **Создание подстановки** выберем опцию **Объект «столбец подстановки» будет использовать значения из таблицы или запроса** и нажмем кнопку **Далее**. В следующем открывшемся окне **Создание подстановки** выберем таблицу, из которой нужно взять значение поля (в данном случае это таблица «ПОКУПАТЕЛИ»), и нажатием кнопки **Далее** переходим к следующему окну (рисунок 2.9).

В этом окне из списка **Доступные поля** выбираем нужное («Клиент»), щелчком на кнопке  переводим его в раздел **Выбранные поля** и снова нажимаем кнопку **Далее**, после чего в появившемся окне остается нажать кнопку **Готово**.

Проделав аналогичные действия с полем «Поставщик» таблицы «ТОВАРЫ» (значения для списка выбираем из таблицы «ПОСТАВЩИКИ»), получим список для ввода значений в это поле.

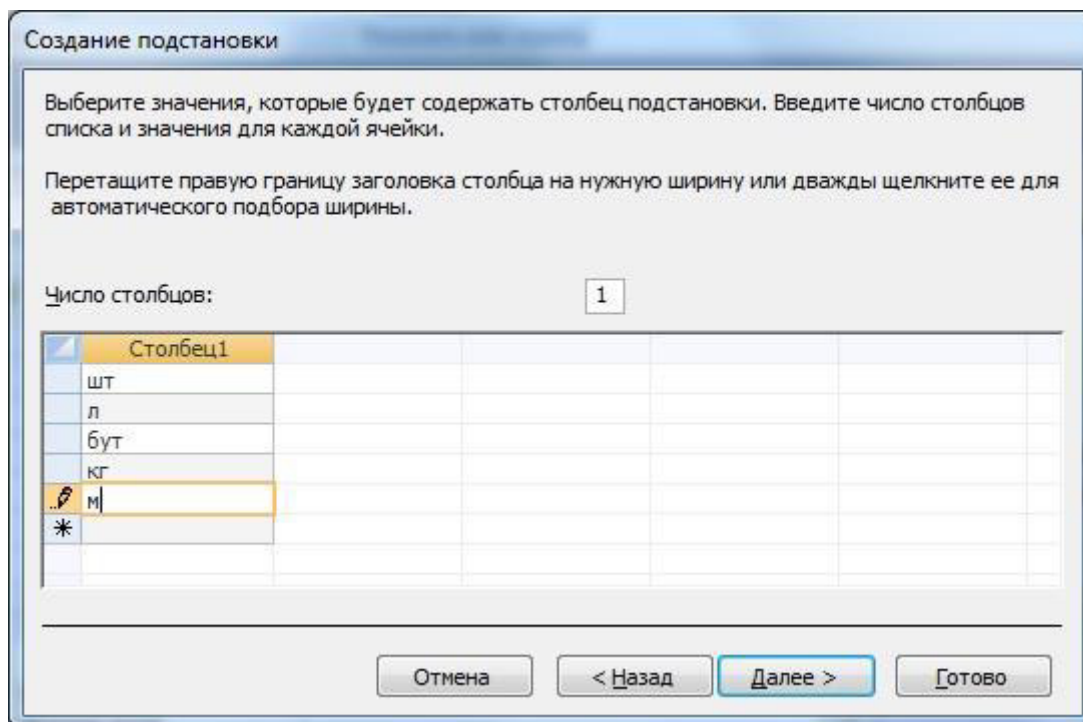


Рисунок 2.8 - Создание фиксированного списка значений поля «Ед. изм.»

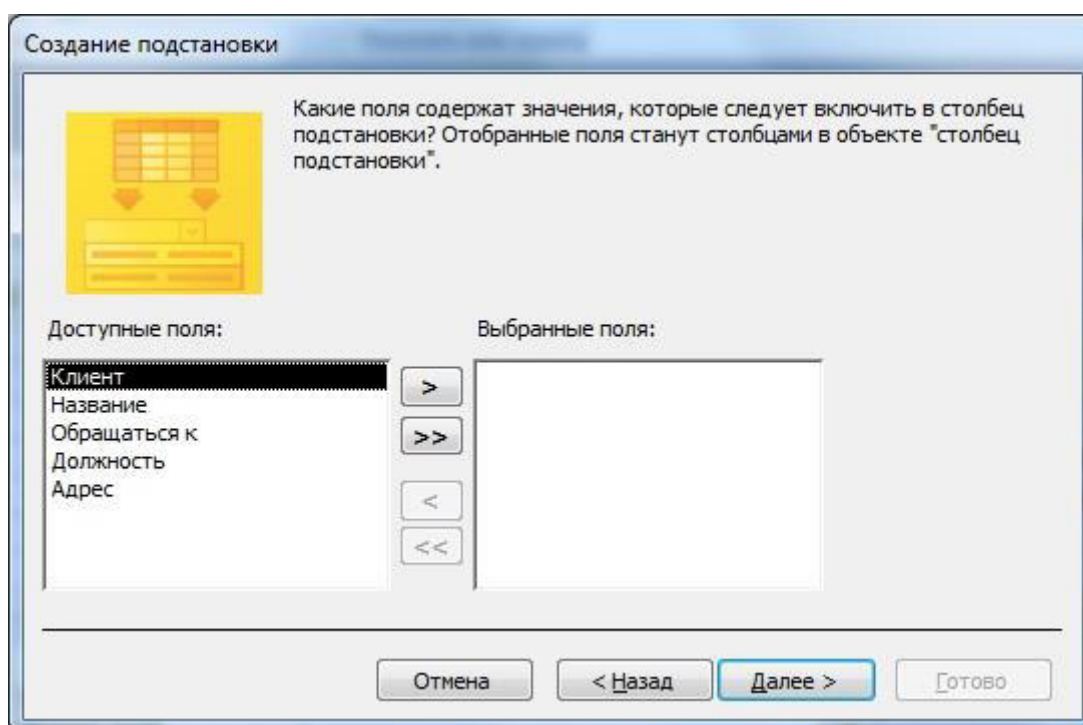


Рисунок 2.9 - Выбор полей для формирования списка

2.6 Связывание таблиц

ACCESS позволяет строить реляционные БД, отдельные таблицы которых могут быть связаны между собой. Как правило, связывают ключевое поле одной таблицы с соответствующим ему полем другой таблицы, которое называют полем внешнего ключа (поля «Поставщик» и «Клиент» таблицы «ТОВАРЫ»).

Связанные поля могут иметь различные имена, но у них обязательно должны быть одинаковые типы данных, одинаковый размер и некоторые другие значения свойств.

В нашем примере необходимо установить связь между таблицами «ТОВАРЫ» (поле «Клиент») и «ПОКУПАТЕЛИ» (поле «Клиент»), а также между таблицами «ТОВАРЫ» (поле «Поставщик») и «ПОСТАВЩИКИ» (поле «Поставщик»).

Перед началом установления связей необходимо закрыть все таблицы, между которыми эти связи будут устанавливаться. Затем на вкладке **Работа с базами данных** нужно выбрать команду **Схема данных**, после чего на экране появится окно **Схема данных**, в котором можно просмотреть и задать связи между таблицами.

Чтобы добавить таблицы в окно **Схема данных**, нужно щёлкнуть по кнопке **Отобразить таблицу** на вкладке **Работа со связями\ Конструктор** группа **Связи**, или, вызвав щелчком правой кнопки мыши контекстное меню, выбрать из него пункт **Добавить таблицу**.

В списке таблиц, находящемся на вкладке **Таблицы** диалогового окна **Добавление таблицы** (рисунок 2.10), необходимо маркировать таблицу «ТОВАРЫ» и нажать кнопку **Добавить**.

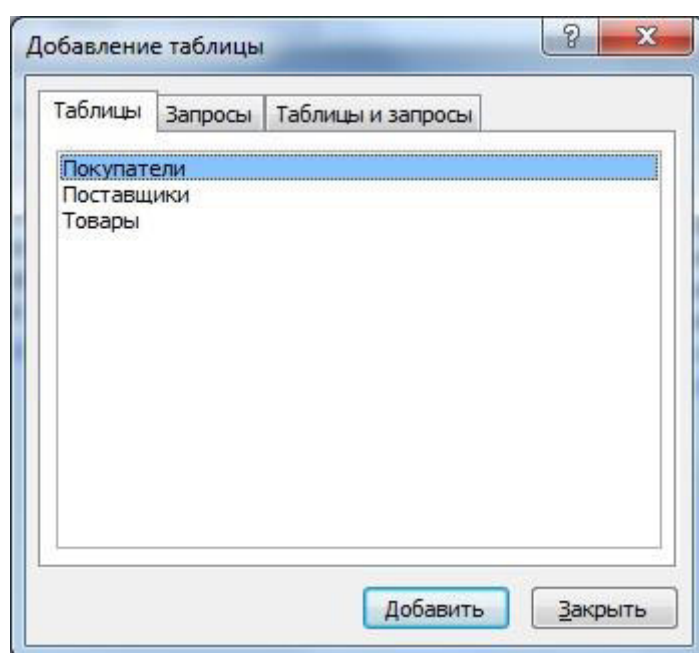


Рисунок 2.10 - Окно **Добавление таблицы**

Те же самые операции необходимо выполнить для таблиц «ПОСТАВЩИКИ» и «ПОКУПАТЕЛИ», после чего в окне **Схема данных** окажутся все три созданные нами таблицы.

Для создания связи нужно с помощью буксировки мышью просто переместить используемое для связи поле (в нашем случае это поле «Клиент» таблицы «ПОКУПАТЕЛИ» и поле «Поставщик» таблицы «ПОСТАВЩИКИ») из одной таблицы к соответствующим полям другой таблицы (в рассматриваемом примере это поля «Клиент» и «Поставщик» таблицы «ТОВАРЫ»).

На экране для каждой связи появится диалоговое окно **Изменение связей** (рисунок 2.11), в котором будет предложена связь между выбранными таблицами. Теперь путем установления типа отношений между таблицами следует определить параметры связи. Активируйте опцию **Обеспечение целостности данных**, что позволит избежать ряда ошибок, допускаемых при удалении записей из первичной таблицы и вводе информации в связанную таблицу.

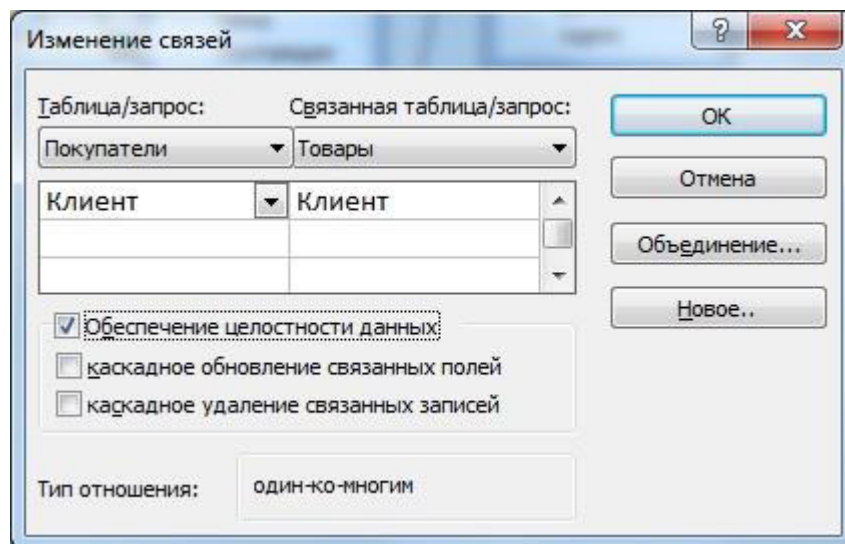


Рисунок 2.11 - Окно задания отношения между таблицами

Для наших таблиц по умолчанию установлено отношение типа «один-ко-многим». Эта установка отображается в поле **Тип отношения**. Таким образом, одной записи таблиц «ПОСТАВЩИКИ» и «ПОКУПАТЕЛИ» может быть поставлено в соответствие несколько записей связанной таблицы «ТОВАРЫ». Отношения такого типа являются наиболее распространенными в реляционных базах данных.

После нажатия кнопки **ОК** в окне **Изменение связей** созданные связи отображаются графически (рисунок 2.12)

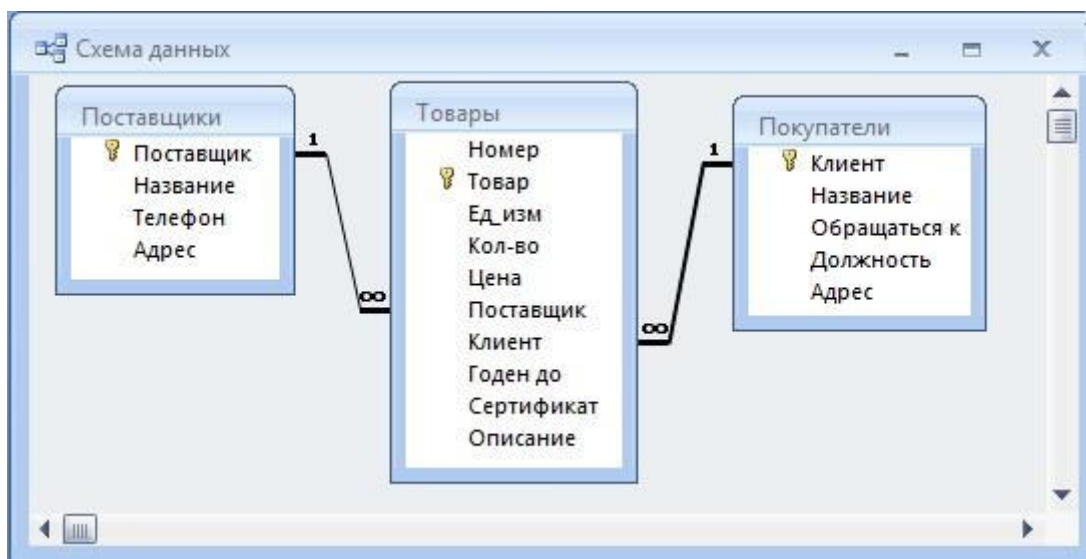


Рисунок 2.12 - Окно **Схема данных** после связывания таблиц

Теперь можно полностью заполнить данными таблицу «ТОВАРЫ». Пример заполненной таблицы показан на рисунке 2.13.

Товары								
№	Товар	Ед. изм.	Кол-во	Цена	Поставщик	Покупатель	Годен до	Сертификат
1	Красное бордо	бут	24	33,50р.	Delikate	Абзац	20.12.2018	☒
2	Индийский чай	кг	50	124,00р.	Тибет	Лебедь	01.01.2019	☒
3	Анисовый сироп	л	30	64,20р.	Fruct	Абзац	31.12.2015	☒
4	Паста из бамии	кг	120	320,00р.	Selderey	Амбра	20.06.2020	☐
5	Тибетское пиво	бут	10	180,00р.	Тибет	Лебедь	01.01.2015	☒
6	Слоновые хвосты	м	22	999,00р.	Тибет	Пиран	10.10.2016	☒
7	Сушёные улитки	шт	100	25,00р.	Delikate	Баран	24.12.2017	☐
8	Клюквенная подлива	кг	36	225,00р.	Selderey	Лебедь	10.12.2017	☒
9	Икра из арбузов	кг	15	360,00р.	Delikate	Лебедь	01.01.2017	☐
10	Испанский сыр	кг	3	189,90р.	Восток	Амбра	02.10.2017	☐
11	Заливное из суши	кг	5	680,00р.	Tokyo	Баран	12.12.2016	☒
12	Ананасы солёные	шт	20	50,00р.	Fruct	Абзац	31.12.2015	☐
13	Лягушачьи лапки	кг	80	280,00р.	Delikate	Пиран	30.01.2018	☒
14	Солёные огурцы	шт	144	3,00р.	Fruct	Лебедь	16.06.2019	☒
15	Варенье из нерпы	л	60	78,60р.	Тибет	Абзац	10.01.2020	☐
16	Кулебяка с фейхуа	кг	60	99,90р.	Fruct	Амбра	20.12.2015	☐
17	Апельсиновый сок	бут	20	33,30р.	Fruct	Баран	26.04.2016	☐
18	Йогурт от лам	л	60	24,40р.	Тибет	Амбра	10.10.2018	☒
19	Сушёная говядина	кг	10	120,00р.	Восток	Пиран	20.02.2017	☐
20	Перепелиные язычки	шт	400	10,00р.	Восток	Абзац	12.02.2016	☒
* (№)			0	0,00р.				☐

Запись: 1 из 20 Нет фильтра Поиск

Рисунок 2.13 - Заполненная таблица «ТОВАРЫ»

3 ЗАПРОСЫ И ФИЛЬТРЫ

Запросы предназначены для отбора данных, удовлетворяющих заданным критериям. Результатом выполнения запроса является набор записей, собранных во временном наборе данных и представленных в виде таблицы.

В ACCESS запросы делятся на QBE – запросы (Query By Example – запрос по образцу), параметры которых устанавливаются в окне конструктора запросов, и SQL – запросы (Structured Query Language – структурированный язык запросов), при создании которых применяются операторы и функции языка SQL. Последний вид запросов чаще всего используется в том случае, если пользователь из ACCESS делает запрос к внешней базе данных, поэтому в данном пособии SQL – запросы не рассматриваются.

3.1 Запросы на выборку

Одним из наиболее распространенных типов QBE – запросов является запрос на выборку. Чтобы научиться создавать такие запросы, создадим запрос на выборку данных из таблицы «ТОВАРЫ», удовлетворяющий условиям: товар поставлен поставщиком Fruct, срок годности истек до 01.02.2016, количество товара – больше или равно 30 единиц. Полученному запросу присвоим имя «ОТПАВА ОТ FRUCT».

Для создания такого запроса выполним следующую последовательность действий:

 **Создание**  **Конструктор запросов**

 **Таблицы**  **ТОВАРЫ**  **Добавить**  **Заккрыть**

Тем самым мы определили исходные данные, с которыми будет работать запрос, и вызвали окно **конструктора** запросов, которое разделено на две части. В верхней половине находятся окна таблиц со списками полей.

В запрос не обязательно включать все поля выбранных таблиц. Например, в рассматриваемом примере нас интересуют: наименование товара, количество, поставщик и срок годности.

Добавить нужные поля в бланк запроса можно выполняя двойные щелчки по их именам в списке, находящемся в верхней части окна **конструктора**, либо выбирая из раскрывающегося списка в строке **Поле** бланка запроса. После выбора полей окно **конструктора** должно выглядеть так, как показано на рисунке 3.1.

В процессе работы с ACCESS большинство запросов используют только часть полей (как в нашем случае), но иногда требуется включить в запрос все поля таблицы. Это можно выполнить следующими способами:

- ◆ Маркируйте все поля, выполнив двойной щелчок на строке заголовка таблицы, и перетащите их в первую строку бланка запроса. ACCESS автоматически разместит каждое поле таблицы в отдельной колонке;
- ◆ Маркируйте символ * в списке полей таблицы и перетащите его в бланк запроса. В результате все поля будут включены в запрос, но в бланке появится только запись **Имя таблицы**.

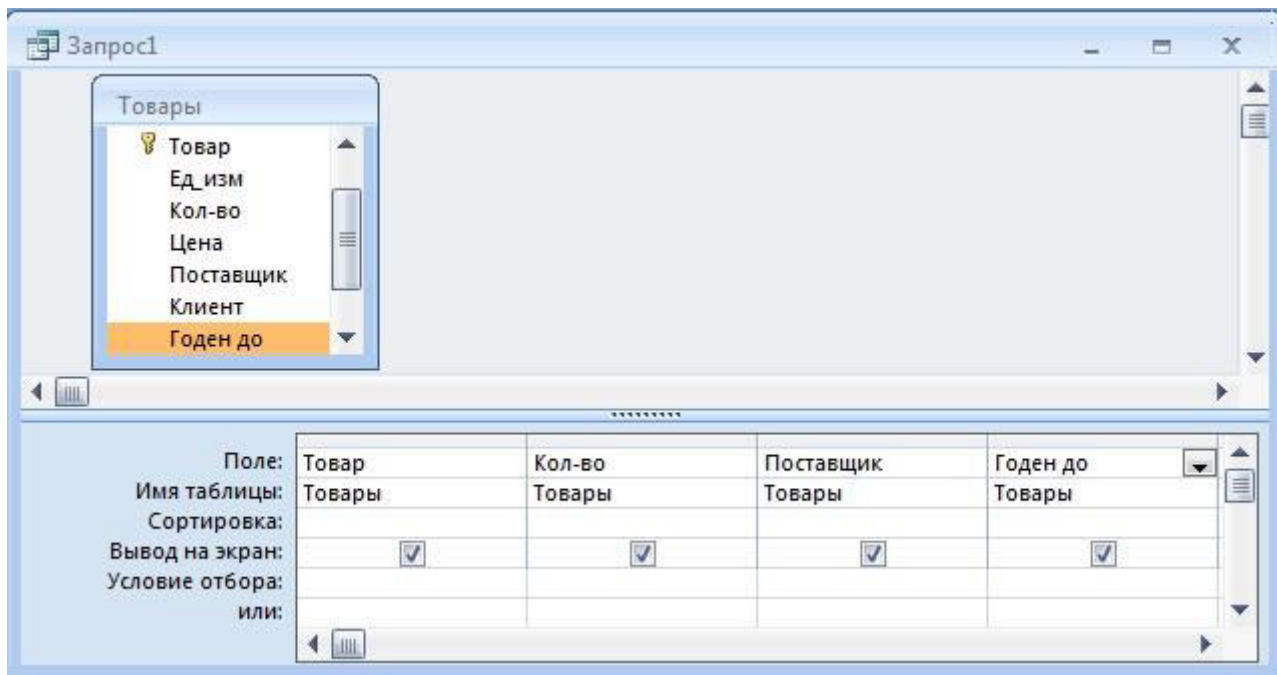


Рисунок 3.1 - Результат включения полей в запрос

Следующим этапом создания запроса является установка критериев отбора записей, которые указываются для одного или нескольких полей. Критериями для нашего отбора будут:

«Поставщик» = Fruct

«Годен до» < 01.02.2016

«Кол-во» >= 30

После ввода каждого критерия нужно нажать клавишу <Enter>, при этом ACCESS проверит синтаксис критерия и нормализует запись в соответствии с внутренними правилами записи критериев.

Записи в генерируемом при выполнении запроса динамическом наборе данных могут быть отсортированы по какому-то полю. Для выполнения сортировки следует перейти в соответствующий столбец бланка запроса и указать способ сортировки в строке **Сортировка**. При выполнении щелчка на этой строке появляется список с перечнем видов сортировки. В нашем примере расположим товары в алфавитном порядке. Для этого в столбце **Товар** выполним щелчок в строке **Сортировка** и в появившемся списке выберем опцию **по возрастанию**.

Окончательно нижняя часть окна запроса представлена на рисунке 3.2.

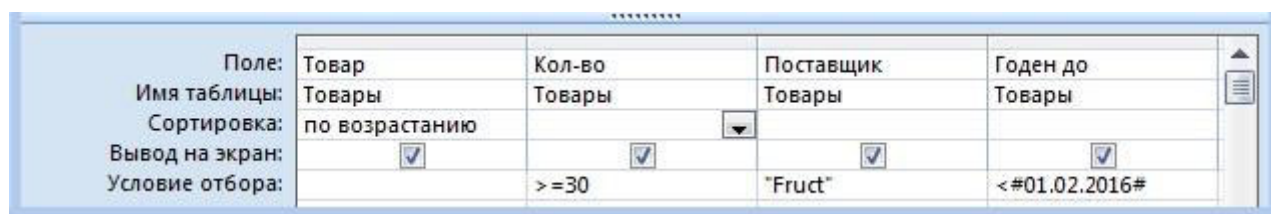


Рисунок 3.2 - Пример задания критерия отбора записей

Осталось, как обычно, сохранить полученный запрос под нужным именем:

➤ **файл** ➤ **Сохранить**

Имя запроса : = Отрава от Fruct

☐ **ОК**

Результат выполнения запроса показан на рисунке 3.3.

Товар	Кол-во	Поставщик	Годен до
Анисовый сироп	30	Fruct	31.12.2015
Кулебяка с фейхуа	60	Fruct	20.12.2015
*	0		

Рисунок 3.3 - Результат выполнения запроса, показанного на рисунке 3.2

В запросах часто используются подстановочные знаки (таблица 3.1), которые позволяют указать образец поиска в следующих случаях:

- ◆ известна только часть значения;
- ◆ требуется найти значения, начинающиеся с конкретной буквы или соответствующие определенному шаблону.

Таблица 3.1 - Подстановочные знаки для поиска значений

Знак	Пояснение	Пример
*	Соответствует любой последовательности символов, может использоваться в качестве первого или последнего элемента текстовой строки.	* а – любое слово, заканчивающееся буквой а
?	Соответствует одному любому текстовому символу.	т?p – поиск слов тор, тир, тур
[]	Соответствует любому одному символу из заключенных в скобки.	т[ou]р – поиск слов тор, тур, но не тир
-	Соответствует любому символу из диапазона.	т[a-o]р – поиск слов тир, тор, но не тур
#	Соответствует любой цифре	1#3 поиск записей 103, 113, 123, ...

В том случае, если в общий критерий поиска требуется вставить оператор ИЛИ, нужно просто использовать новую строку для каждого условия.

На рисунке 3.4 показан пример более сложного запроса поиска записей из таблицы «ТОВАРЫ», удовлетворяющих условиям:

- ◆ название товара начинается с буквы А или заканчивается буквой и;
- ◆ единица измерения товара – кг;
- ◆ количество товара, начинающегося с буквы А, находится в диапазоне [20-80];
- ◆ цена товара, заканчивающегося буквой и, либо меньше 100 рублей, либо больше 400 рублей.

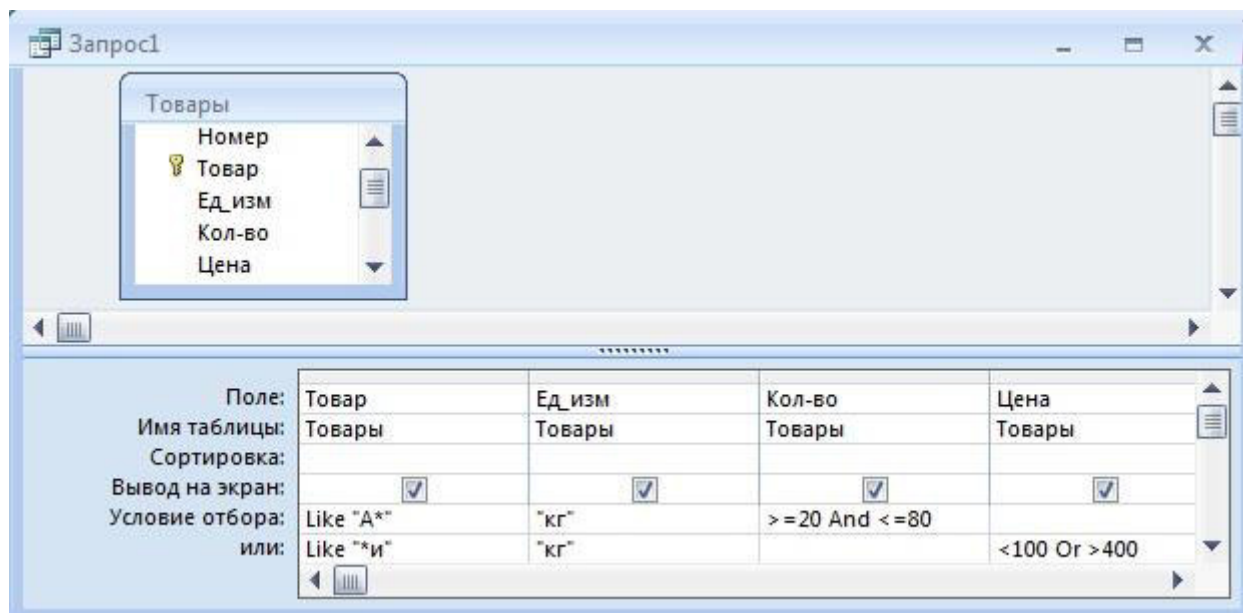


Рисунок 3.4 - Пример сложного запроса на выборку

3.2 Запросы на удаление

Запросы этого типа служат для удаления из таблицы групп записей, соответствующих некоторому критерию отбора.

Прежде всего необходимо разработать запрос на выборку удаляемых записей, как это было рассмотрено в предыдущем разделе, желательно, выполнить этот запрос, чтобы проверить корректность заданных критериев. Затем следует вернуться в режим конструктора и выбрать в группе **Тип запроса** команду **Удаление**. ACCESS добавит в бланк запроса строку **Удаление** и введет в ее ячейки значение **Условие**. Это означает, что пользователь может установить дополнительные критерии отбора. Чтобы выполнить запрос щёлкните по кнопке **Выполнить** (с изображением восклицательного знака) в группе **Результаты** на вкладке **Конструктор**.

Перед выполнением запроса на удаление ACCESS дважды потребует подтверждения выполнения операции. Первое подтверждение (рисунок 3.5) появляется при любом действии, связанном с возможностью потери или изменения информации в базе данных, никакого анализа запроса при этом не производится. После нажатия на кнопку **Да** запрос будет проанализирован и в специальном диалоговом окне (рисунок 3.6) ACCESS укажет, сколько записей удаляется из таблицы и потребует подтвердить удаление. Если запрос составлен с ошибкой, то в окне, показанном на рисунке 3.6 вместо числа удаляемых записей будет сообщение об ошибке в запросе.

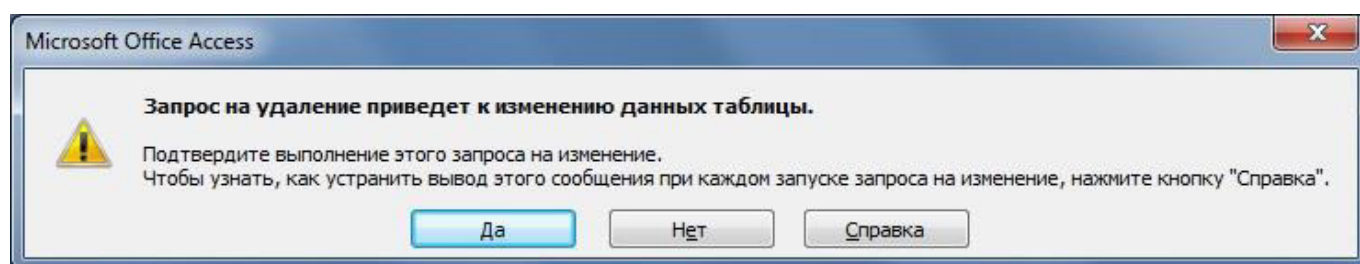


Рисунок 3.5 - Первое диалоговое окно запроса на удаление.

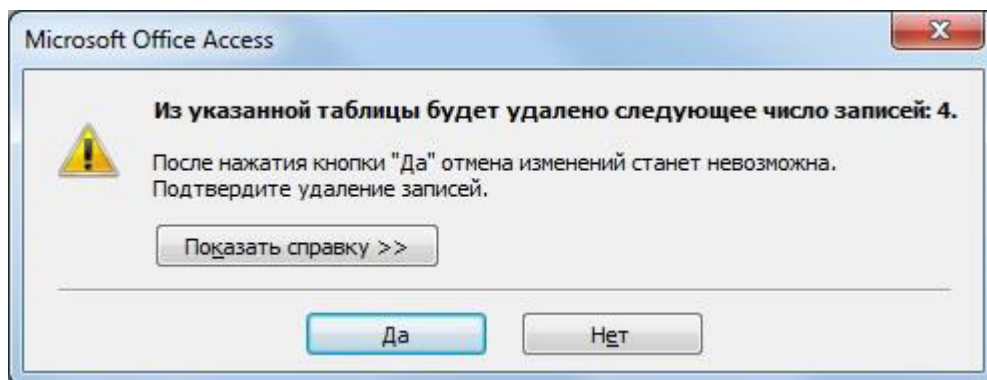


Рисунок 3.6 - Диалоговое окно с указанием числа удаляемых записей.

Если в окне исходных таблиц для запроса (верхняя часть окна конструктора запросов) присутствует несколько связанных таблиц, то в запросе необходимо обязательно указать из какой таблицы нужно удалять записи. Для этого необходимо в соответствующей таблице в списке полей выполнить двойной щелчок по строке, содержащей звездочку, при этом в нижней части окна конструктора запроса в строке **Удаление** появится значение **Из**.

Записи, удаленные посредством запроса, нельзя восстановить, поэтому следует тщательно анализировать критерии отбора.

3.3 Запрос на обновление

Используя запрос на обновление, пользователь может изменить группу записей, отобранную на основе определенных критериев. При разработке запроса на обновление прежде всего создается и проверяется запрос на выборку. Затем в режиме конструктора активизируется команда **Обновление** в группе **Тип запроса**, после чего ACCESS добавляет в бланк запроса строку **Обновление**, которая предназначена для указания новых значений полей таблицы. После нажатия кнопки **Выполнить** ACCESS укажет в специальном диалоговом окне, сколько записей изменится в таблице и потребует подтвердить изменения.

На рисунке 3.7 показан пример запроса на обновление таблицы “ТОВАРЫ”, заключающийся в том, что товарам со сроком годности до 01.01.2016 ставится в соответствие отсутствие сертификата (значение поля – *нет*).

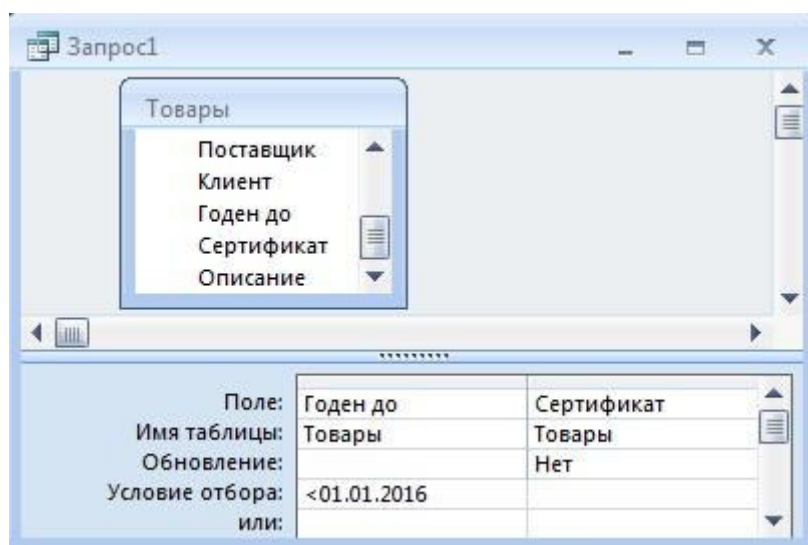


Рисунок 3.7 - Пример запроса на обновление таблицы

3.4 Создание вычисляемых полей в запросе

В таблице «ТОВАРЫ» есть сведения об объеме каждой партии товара (поле «Кол-во») и цене единицы товара (поле «Цена»). Можно предположить, что в реальной задаче потребуется создать поле, в котором была бы указана стоимость каждой партии товара. Для этого в запросе нужно создать вычисляемое поле.

Создадим запрос, содержащий сведения о товарах, поставляемых фирмой Fruct и содержащий общую стоимость каждой партии товара.

Вначале создадим обычный запрос на товары от фирмы Fruct, а затем в свободной графе бланка в строке **Поле** введем выражение, которое определит значение вычисляемого поля:

[ТОВАРЫ].[Кол-во]*[ТОВАРЫ].[Цена]

Готовый запрос показан на рисунке 3.8, результат выполнения - на рисунке 3.9.

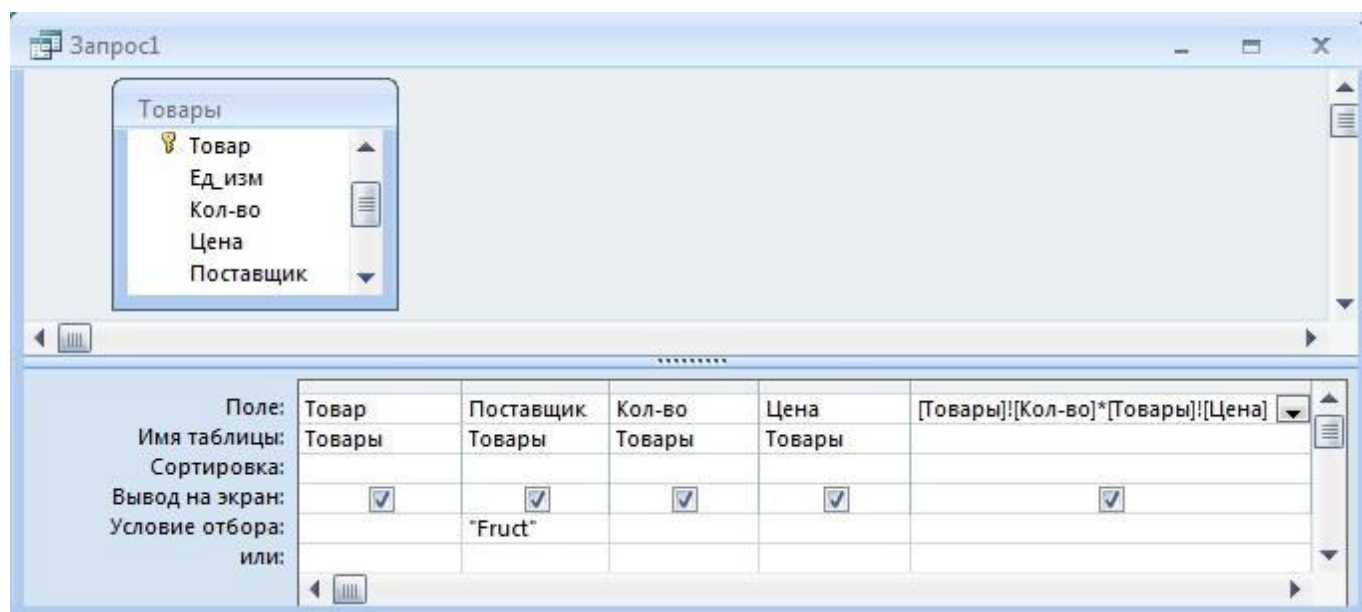


Рисунок 3.8 - Создание вычисляемого поля в запросе

Товар	Поставщик	Кол-во	Цена	Выражение
Анисовый сироп	Fruct	30	64,20р.	1 926,00р.
Ананасы солёные	Fruct	20	50,00р.	1 000,00р.
Солёные огурцы	Fruct	144	3,00р.	432,00р.
Кулебяка с фейхуа	Fruct	60	99,90р.	5 994,00р.
Апельсиновый сок	Fruct	20	33,30р.	666,00р.
*		0	0,00р.	

Рисунок 3.9 - Запрос с вычисляемым полем

Как видно из приведенного примера, имена объектов и полей заключаются в квадратные скобки, имя объекта базы данных отделяется от имени поля восклицательным знаком.

Имя вычисляемого поля ACCESS назначил сам, если вам не нравится заголовок колонки “Выражение1”, то есть два способа изменить его. Можно задать другое имя прямо в бланке запроса или выбрать из контекстного меню этого поля команду **Свойства** и там определить подпись.

3.5 Построитель выражений

Формировать условия отбора и выражения в запросах удобно с помощью построителя выражений, показанного на рисунке 3.10. Он запускается командой **Построить** из контекстного меню, связанного со строкой **Поле** (для создания вычисляемого поля) либо со строкой **Условие отбора** (для задания выражения в условии отбора) на бланке запроса.

Окно построителя выражений имеет четыре области со своими полосами прокрутки.

В верхней области располагается создаваемое выражение. Имена объектов и полей заключаются в квадратные скобки, символьные строки – в кавычки, даты – в знаки #. Имя объекта базы данных отделяется от имени поля восклицательным знаком.

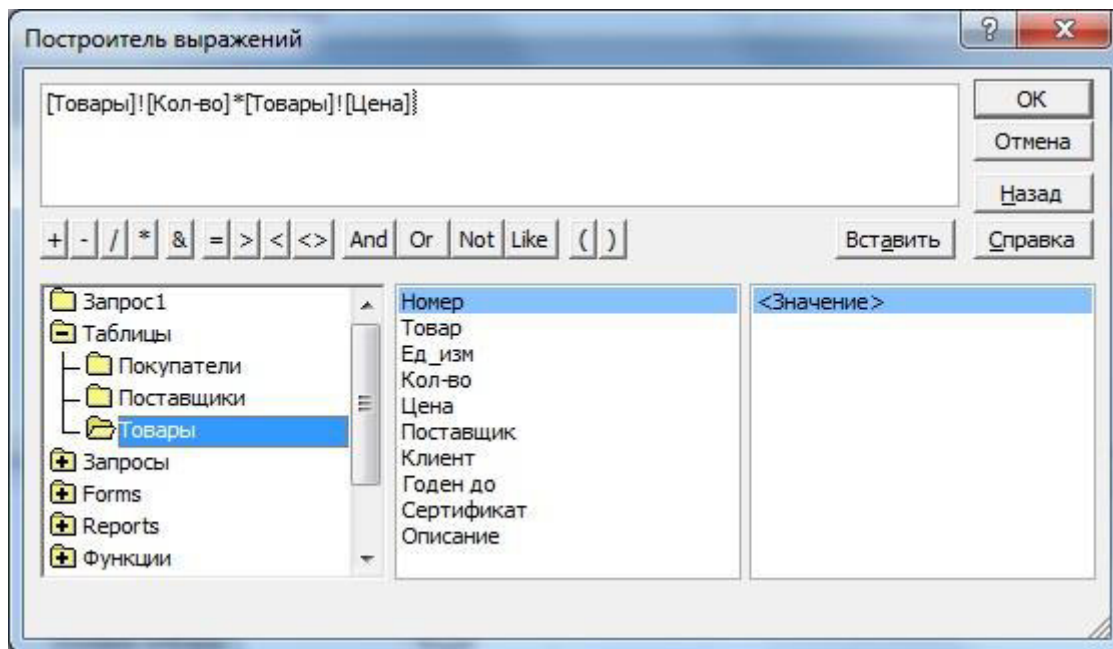


Рисунок 3.10 - Окно построителя выражений

Три нижние области используются для выбора элементов запроса.

Левая область содержит список всех источников данных для построения выражения. Как видно из рисунка 3.10, ими могут быть объекты БД: таблицы, запросы, формы и т.д. Кроме этого, в левый список обычно включаются папки, содержащие функции, константы и операторы, используемые в выражении.

Средний список служит для показа элементов, входящих в выбранный объект из левого списка, например, если в левой области указано имя конкретной таблицы, то средний список будет содержать имена ее полей.

Правый список служит для отображения и выбора самых примитивных объектов (функций, текста, чисел и т.д.).

Для того чтобы вставить имя объекта в выражение, нужно выполнить двойной щелчок на соответствующем объекте в одном из нижних списков.

Кнопка **Назад** позволяет отменить ввод данных в выражение.

Кнопки с изображенными на них знаками математических действий позволяют быстро вводить в выражение соответствующие символы.

Кроме традиционных математических действий существуют еще несколько операторов:

BetweenAnd – заменяет знаки $\geq$ и $\leq$. Например, условие

Between 1981 And 1994 эквивалентно условию ≥ 1981 And ≤ 1994 .

Оператор In позволяет использовать списки значений. Например, выражение In(“Fruct”;”Тибет”) позволяет отобрать записи с указанными значениями.

^ - обозначает операцию возведения в степень.

Or, And – операции логического сложения и умножения.

3.6 Групповые операции в запросе

Запрос можно использовать для выполнения расчетов и подведения итогов, обобщив данные из исходных таблиц. Для этих целей предусмотрены статистические функции, которые задают в строке **Групповая операция** запроса. Данная строка появляется после нажатия на кнопку **Итоги** в группе **Показать или скрыть** вкладки **Конструктор**. В таблице 3.2 приведены некоторые элементы раскрывающегося списка в ячейке строки **Групповая операция**.

Таблица 3.2 - Элементы списка **Групповая операция**

Элемент списка	Результат
Sum	Сумма значений поля записей, отобранных запросом
Avg	Среднее значение полей в записях, отобранных запросом
Min	Наименьшее значение поля в записях, отобранных запросом
Max	Наибольшее значение поля в записях, отобранных запросом
Count	Число значений записей, отобранных запросом
Группировка	Определяет группы, для которых выполняются вычисления

Рассмотрим несколько примеров использования групповых операций.

3.6.1 Вычисление итоговых значений для всех записей

Предположим, что нам нужно вычислить количество видов товаров на складе, измеряемых в штуках и среднюю цену этих товаров. Вид такого запроса показан на рисунке 3.11, а результат его выполнения – на рисунке 3.12.

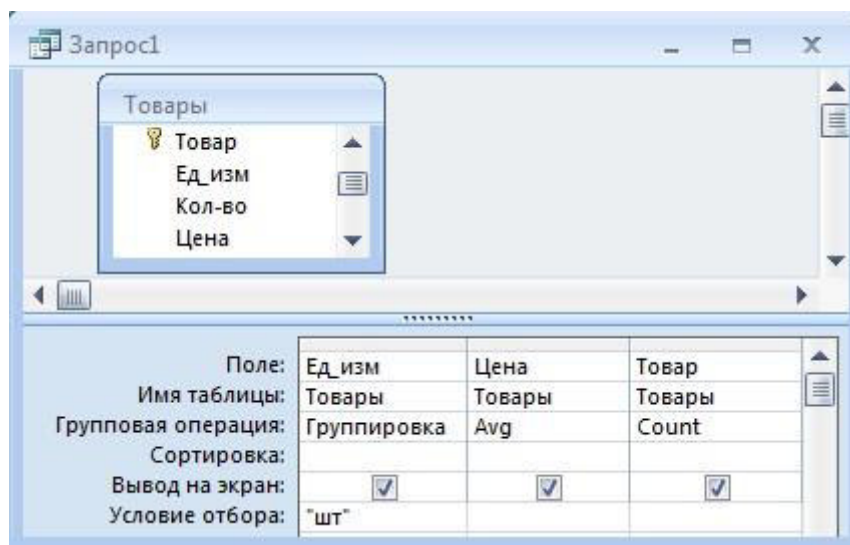


Рисунок 3.11 - Пример 1 запроса с вычислением итоговых значений

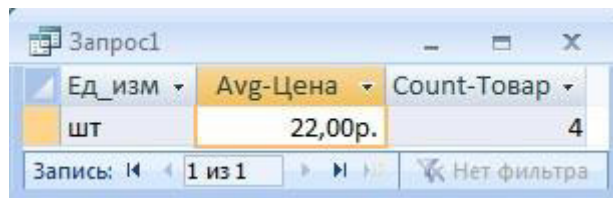


Рисунок 3.12 - Результат выполнения запроса примера 1

3.6.2 Вычисление итоговых значений для групп записей в запросе

Получим распределение общей стоимости товаров, полученных от различных поставщиков, по видам поставщиков. Такой запрос показан на рисунке 3.13, а соответствующий результат – на рисунке 3.14.

Если вычисляемое поле определяется с помощью статистической функции, то ACCESS создает имя, под которым поле будет выводиться в режиме таблицы путем объединения имени функции и имени поля, содержащего данные (например, Avg_Цена на рисунке 3.12). Для изменения имени можно воспользоваться способами, описанными в разделе 3.4.

Если вычисляемое поле определяется с помощью выражения, которое включает несколько статистических функций, необходимо выбрать в ячейке строки **Групповая операция** элемент **Выражение**.

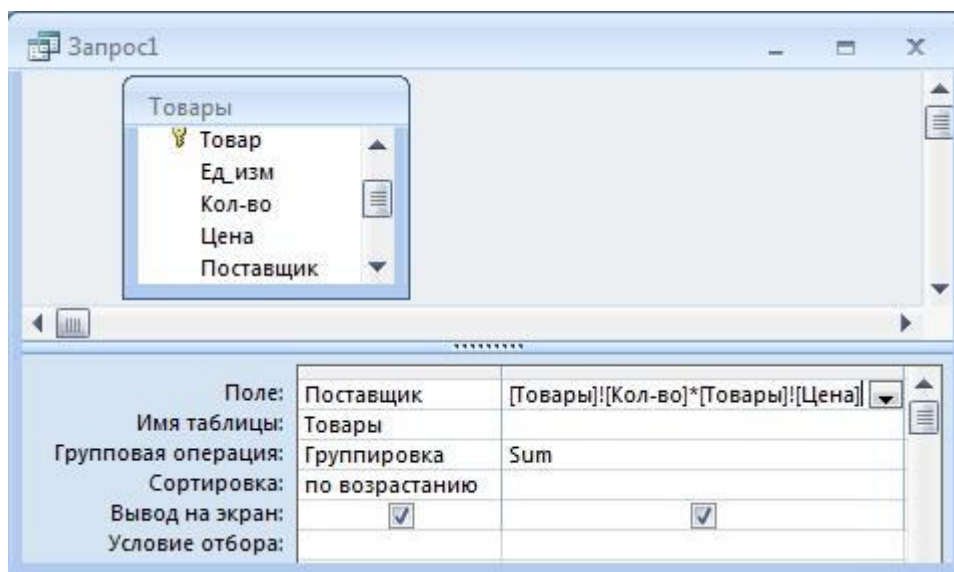


Рисунок 3.13 - Пример 2 запроса с вычислением итоговых значений

Поставщик	Выражение
Delikate	31 104,00р.
Fruct	10 018,00р.
Selderey	46 500,00р.
Tokyo	3 400,00р.
Восток	5 769,70р.
Тибет	36 158,00р.

Запись: 1 из 6

Рисунок 3.14 - Результат выполнения запроса примера 2

3.7 Фильтры

По назначению фильтр представляет собой упрощенный вариант запроса. Если запрос является важнейшим объектом БД, то фильтр – это процедура поиска и сортировки данных, но и фильтры, и запросы выполняют одинаковую функцию – отбор данных.

Существуют следующие виды фильтров:

- ◆ фильтр по выделенному;
- ◆ обычный фильтр;
- ◆ расширенный фильтр.

Вызываются фильтры либо из контекстных меню, связанных с заголовком окна и с ячейками таблицы, либо из группы **Сортировка и фильтр** на вкладке **Главная**.

3.7.1 Фильтр по выделенному

Критерием фильтрации служит ячейка, в которой находится курсор в таблице или запросе. Чтобы воспользоваться этим типом фильтрации следует, например, выделить в таблице “ТОВАРЫ” ячейку с именем какого-либо поставщика, выполнить команду **Выделение** и выбрать необходимое действие (рисунок 3.15).

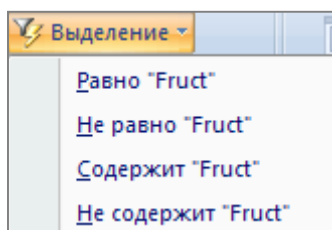


Рисунок 3.15–Список действий при выполнении команды **Выделение**

Если выбрать **Равно**, в данном примере “**Fruct**”, то в таблице останутся видимыми только те записи, в которых данное поле содержит выделенное значение.

Выбор команды **Не равно**, в данном примере “**Fruct**”, обеспечивает обратную фильтрацию данных – скрываются записи, значения полей которых совпадают со значениями выделенных полей.

Команда **Удалить фильтр** не удаляет сам использованный критерий, а только отменяет действие фильтра, т.е. опять все записи становятся доступными. Включить режим фильтрации снова можно командой **Применить фильтр**.

3.7.2 Обычный фильтр

Данный фильтр запускается командой

Главная/ Сортировка и фильтр/ Дополнительно/Изменить фильтр

и позволяет отредактировать условие отбора записей в окне обычного фильтра. По этой команде открывается окно, в котором критерий фильтрации по каждому полю представлен в виде списка. Список можно раскрыть и изменить набор значений полей, которые первоначально не использовались для задания критерия фильтрации. Пример такого окна показан на рисунке 3.16. Фильтр создан для запроса о товарах для покупателя **Амбра** с единицей измерения товара – **шт**. Критерий фильтрации можно определить для любого поля, используя соответствующий список.

Вкладка **или** позволяет создать дополнительные критерии. Объединение по **или** в данном случае означает, что после фильтрации будут показаны совместно все записи, которые были отобраны по критериям на каждой из вкладок.

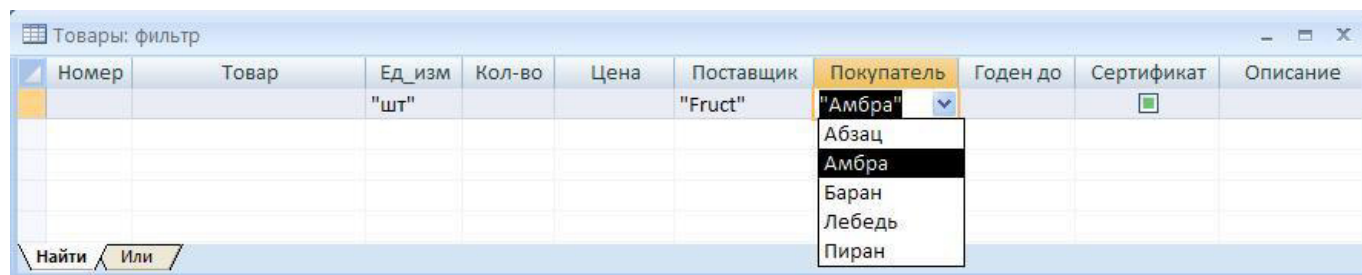


Рисунок 3.16 - Окно обычного фильтра

3.7.3 Расширенный фильтр

Расширенный фильтр, окно которого показано на рисунке 3.17, запускается командой: **Главная/ Сортировка и фильтр/ Дополнительно/Расширенный фильтр**

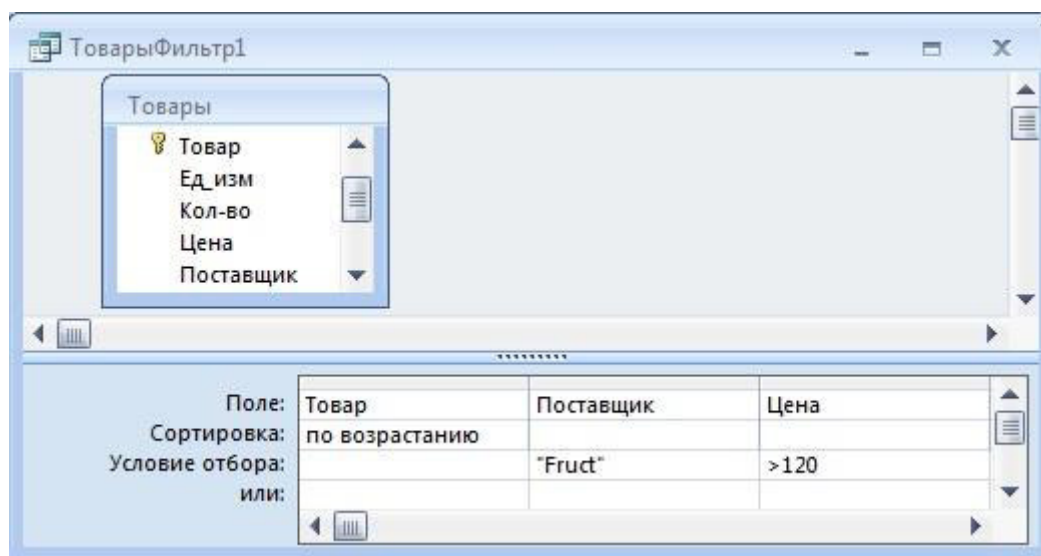


Рисунок 3.17 - Окно расширенного фильтра

На рисунке 3.17 расширенный фильтр используется для поиска товаров, поставляемых поставщиком Fruct с ценой выше 120 рублей. Окно расширенного фильтра похоже на окно запроса в режиме конструктора, основное отличие состоит в отсутствии строки **вывод на экран**.

Расширенный фильтр может быть сохранён как запрос.

4 РАБОТА С ФОРМАМИ

Данные чаще всего отображаются в ACCESS в виде таблиц, что удобно для одновременного просмотра большого количества записей, но нередко более наглядным является другой способ представления информации – в виде формы, в которой видна только одна запись. Такой способ работы с БД рекомендуется использовать, если вводимые данные будут часто изменяться, поскольку в режиме формы можно сконцентрировать внимание на информации, относящейся к определенной записи. Кроме того, форма позволяет расположить данные на экране так, как это требуется, что позволяет значительно облегчить работу с информацией, особенно для неквалифицированных пользователей ACCESS.

Любая форма строится на основе таблицы либо запроса. Имена полей извлекаются из спецификации таблицы. Поля в форме можно расположить по своему усмотрению, объединив при этом логически связанные поля в группы, а также внести в форму элементы управления. На основе одной таблицы можно построить несколько форм: одну – для руководителя, другую – для работника склада, третью – для бухгалтера и т.д.

В форме вся информация содержится в элементах управления. Некоторые элементы связаны с полями базовой таблицы. В таких элементах можно показывать и изменять содержимое соответствующих полей таблицы. Другие элементы формы, например, названия полей, служат для её оформления.

Форму можно создать с помощью мастера форм либо в режиме конструктора форм.

Мастер форм предлагает самый простой способ создания формы. Задав вам целый ряд вопросов, он выдает довольно примитивную форму. **Конструктор форм** предлагает вам пустую форму и набор инструментов, оперируя которыми вы можете спроектировать любую нужную форму. Часто применяется комбинированный метод: создав вначале простую форму с помощью **мастера**, ее изменяют средствами **конструктора**.

4.1 Мастер форм

Для показа возможностей **Мастера форм** получим форму, содержащую сведения о продуктах из таблицы «ТОВАРЫ» и характеристики покупателей соответствующих товаров, хранящиеся в таблице «ПОКУПАТЕЛИ».

Чтобы запустить **Мастер форм** выполним следующую последовательность действий:

☐ **Создание форм** ☐ **Мастер форм**

На экране появится диалоговое окно **Мастера форм**, показанное на рисунке 4.1.

Здесь надо указать поля, которые необходимы для построения формы, которые можно выбирать из всех имеющихся таблиц и запросов. Отдельные поля можно перенести в список **Выбранные поля** нажатием кнопки , сразу все поля выбираются кнопкой .

Выберем все поля из таблицы «ТОВАРЫ», затем в списке **Таблицы/Запросы** активизируем таблицу «ПОКУПАТЕЛИ» и перенесем в список **Выбранные поля** поля «Название», «Обращаться к», «Должность», «Адрес».

После нажатия кнопки  **Далее** на экране появляется второе окно **Мастера форм**, представленное на рисунке 4.2.

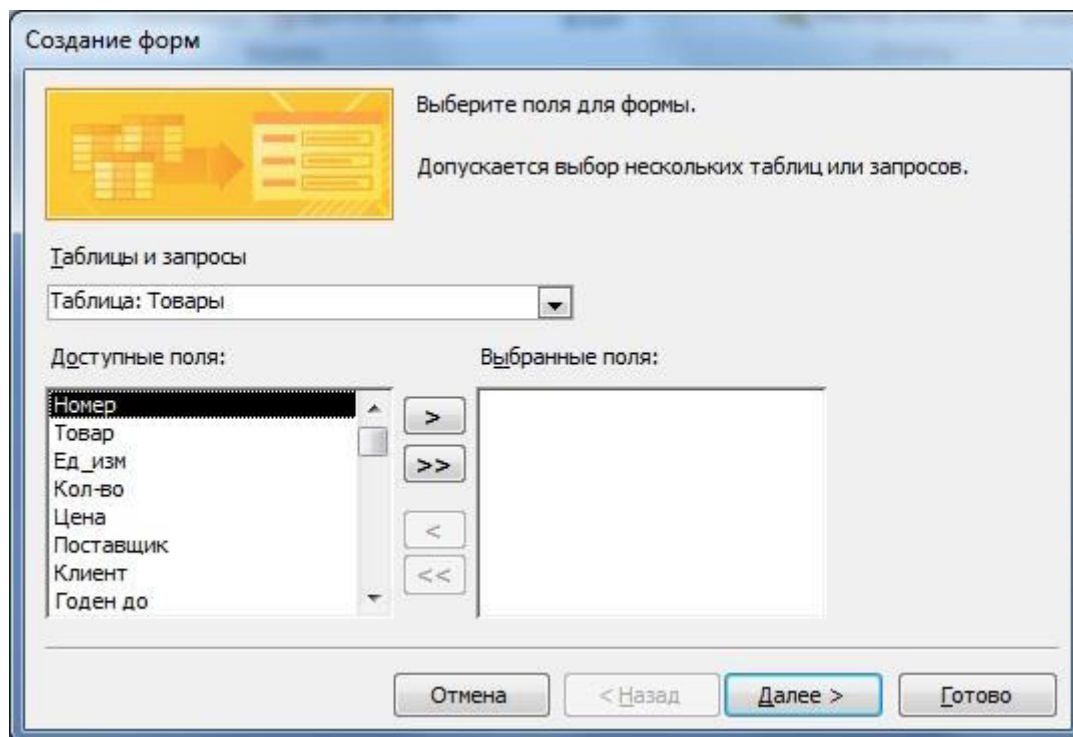


Рисунок 4.1 - Первый шаг **Мастера форм**

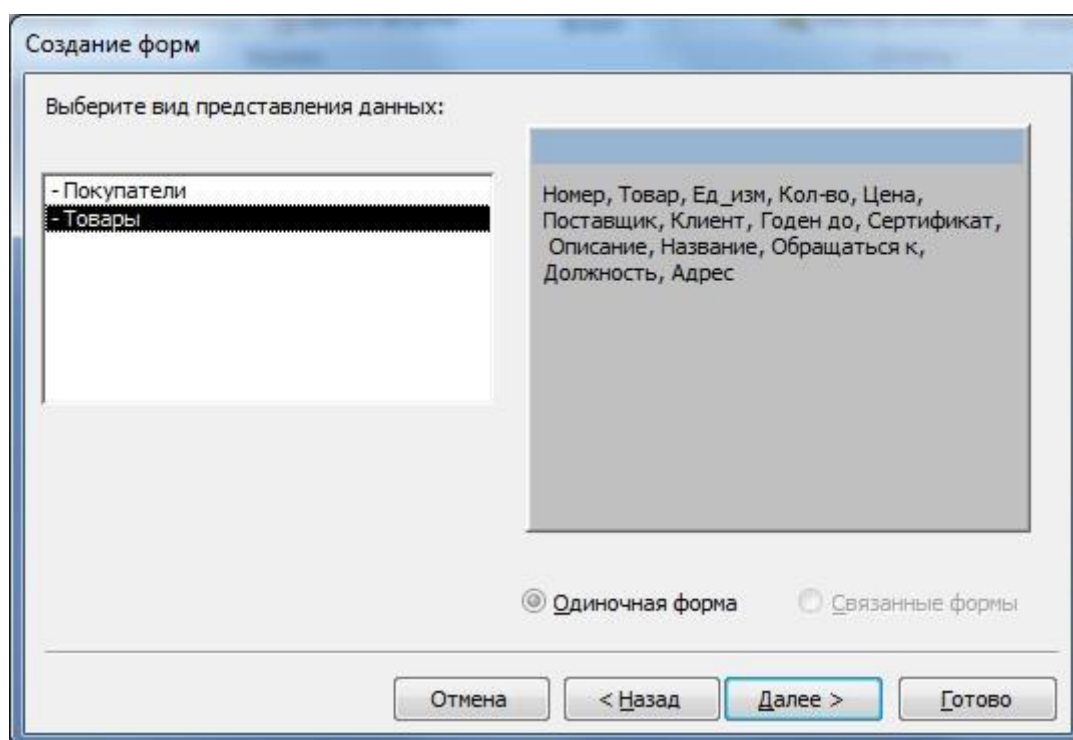


Рисунок 4.2 - Второй шаг **Мастера форм**

Данное окно появляется только в том случае, если в форму включаются поля из нескольких таблиц. В нем необходимо указать (окно **Тип представления данных**), какая из указанных таблиц является основной, справа при этом приводится список полей, содержащихся в проектируемой форме. Выберем в качестве основной таблицу «ТОВАРЫ».

Третий шаг **Мастера форм** показан на рисунке 4.3.

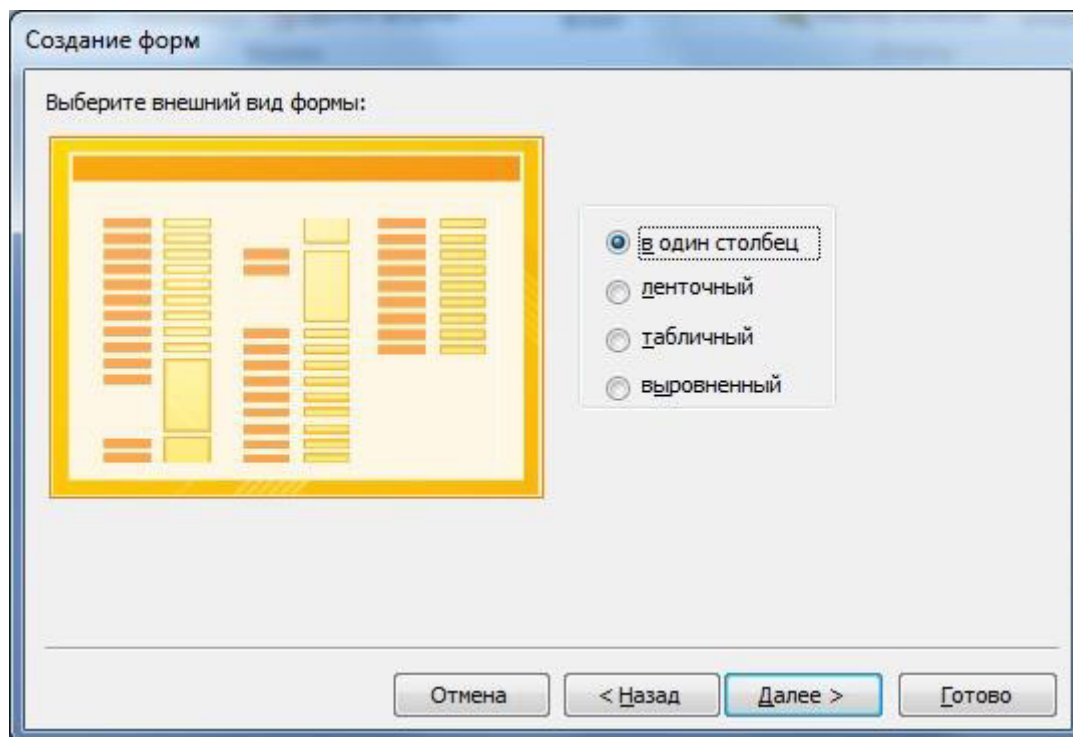


Рисунок 4.3 - Третий шаг **Мастера форм**

Здесь необходимо указать один из приведенных в списке видов формы. В нашем примере выбран вид «в один столбец», который является наиболее удобным для практического использования.

Четвертый шаг проектирования показан на рисунке 4.4. Здесь пользователь может выбрать стиль формы. Это подразумевает создание фона, шрифтов, рельефа подписей. Выбрав стиль, можно сразу увидеть, что он из себя представляет. Картинка в левой части окна напоминает, что **Мастер форм** использует в форме подписи полей, если они были определены, а не имена полей.

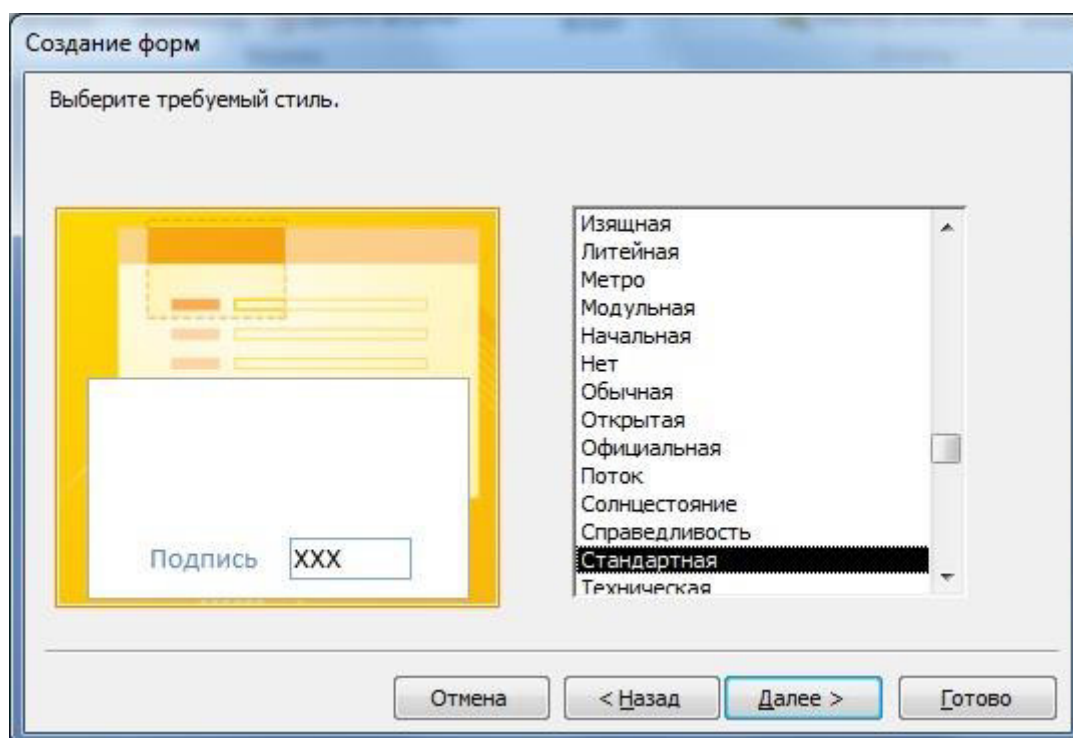


Рисунок 4.4 - Четвертый шаг **Мастера форм**

Последний, пятый шаг **Мастера форм**, показан на рисунке 4.5.

Рисунок 4.5 - Пятый шаг **Мастера форм**

Здесь можно задать имя новой формы и выбрать направление дальнейшей работы: просмотреть готовую форму или доработать то, что не сумел сделать **Мастер форм**.

Результат работы **Мастера форм** показан на рисунке 4.6. Достоинством его работы является быстрота создания формы, но качество дизайна часто не удовлетворяет разработчика.

Рисунок 4.6 - Форма, созданная **Мастером форм**

Например, не все поля и подписи в полученной форме имеют нужный размер - в нашем случае они оказались больше чем нужно. В таких случаях пользователю приходится дорабатывать форму в режиме конструктора.

4.2 Работа с конструктором форм

Откроем форму «ТОВАРЫ», созданную в предыдущем разделе в режиме конструктора, для чего: выполним щелчок правой кнопкой мыши на форме «ТОВАРЫ» в списке объектов базы данных «Склад» и выберем команду **Конструктор**.

Вид получившегося окна показан на рисунке 4.7.

При открытии формы для редактирования в режиме конструктора появляется вкладка **Конструктор**, на которой в группе **Элементы управления** содержатся основные инструменты для работы с формой. Каждая кнопка в этой группе стилизована под элемент управления, который можно встроить в форму. Обзор кнопок группы **Элементы управления** приведен в таблице 4.1.

Таблица 4.1 - Кнопки панели управления **конструктора форм**

Кнопка	Название	Функции
	Надпись	Вставка в форму произвольной записи
	Поле	Включение в форму некоторого поля записи базы данных или вычисляемого поля
	Кнопка	Создание командной кнопки
	Поле со списком	Создание комбинированного списка
	Список	Создание обычного списка
	Подчиненная форма/отчет	Встраивание подчиненной формы в главную форму и установка отношений между ними
	Линия	Проведение в форме разделительной линии
	Прямоугольник	Создание в форме прямоугольной рамки для группы полей
	Присоединенная рамка объекта	Создание рамки объекта, для которого будет установлена связь с файлом-источником
	Группа переключателей	Создание и размещение группы переключателей или флажков
	Флажок	Создание флажка
	Переключатель	Создание переключателя
	Выключатель	Создание выключателя, кнопки с фиксацией
	Набор вкладок	Создание формы или диалогового окна с несколькими вкладками

	Вставить вкладку	Добавление в форму вкладок
	Вставить диаграмму	Отображение данных исходной таблицы или запроса в виде диаграммы
	Свободная рамка объекта	Создание рамки объекта, для которого нельзя установить связь
	Рисунок	Встраивание рисунка (графических файлов) в форму
	Разрыв страницы	Установка принудительного конца страницы формы
	Вставить гиперссылку	Добавление гиперссылки на другие объекты базы данных или адреса Интернет
	Вложение	Вложение файлов и изображений в форму
	Выбор объектов	Выделение объектов (позволяет маркировать и перемещать поля, а также изменять их размеры, установленные по умолчанию)
	Мастера	Включение или отключение мастеров, помогающих создавать элементы управления
	Дополнительные элементы	Встраивание в форму элементов, не представленных на панели инструментов

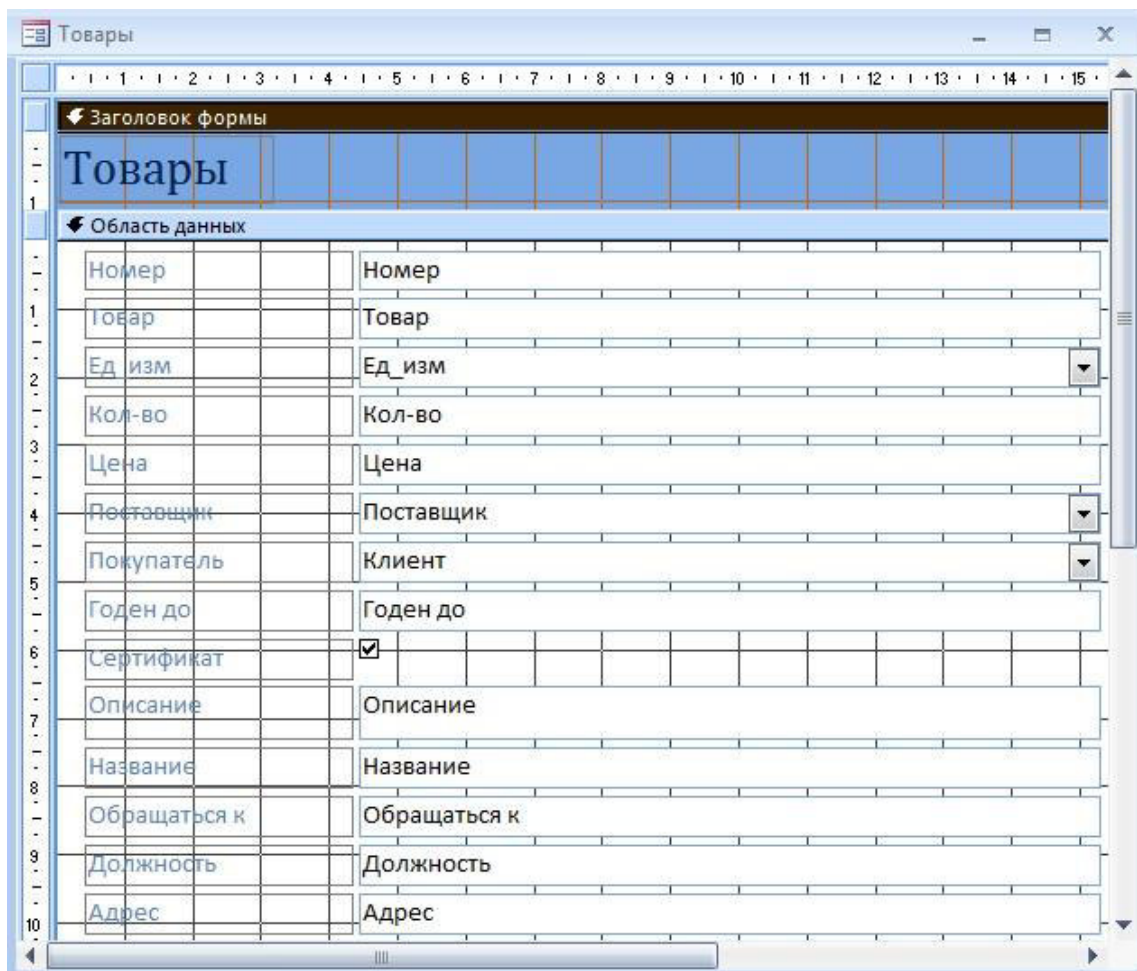


Рисунок 4.7 - Форма «ТОВАРЫ», открытая в режиме конструктора

В области данных, открытой в режиме конструктора формы, находятся поля соответствующих таблиц. Каждое поле состоит в этом режиме из двух элементов: слева – название поля (тип элемента – *Подпись*), справа – содержимое поля (тип элемента *Поле*). Название поля содержит либо текст, определенный в свойстве **Подпись** при проектировании таблицы, либо имя поля.

Изменение размеров формы производится точно так же, как и изменение размеров любого Windows-окна.

4.2.1 Перенос полей и изменение их размеров

При маркировке поля щелчком, оно приобретает вид, показанный на рисунке 4.8, при этом поле можно переносить в любое место формы и изменять его размеры.

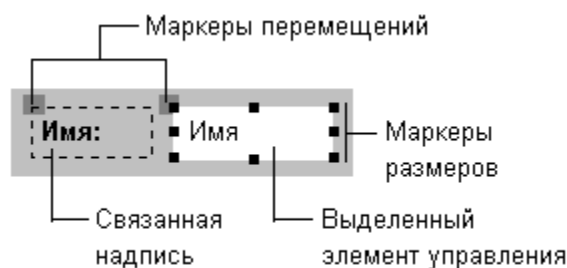


Рисунок 4.8 - Маркированное поле формы

Существует возможность перемещать компоненты полей совместно, если указатель мыши принимает вид кисти руки (рисунок 4.9а) или отдельно друг от друга, если указатель имеет вид кисти с указательным пальцем (рисунок 4.9 б).

Изменить размеры любой компоненты поля можно с помощью перемещения мышью маркеров изменения размеров (рисунок 4.8).

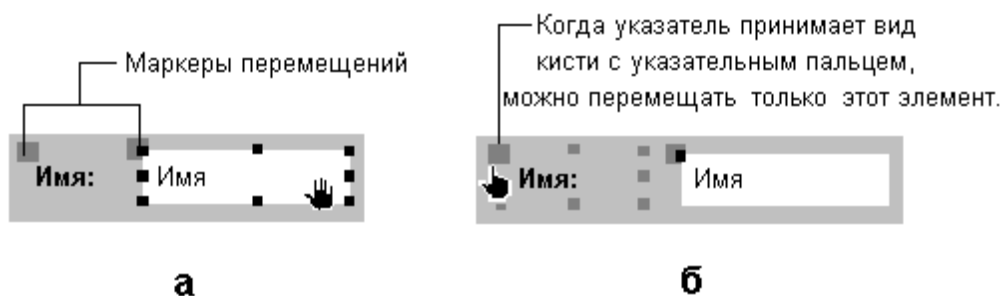


Рисунок 4.9 - Совместный (а) и раздельный (б) перенос компонент полей

Разместим поля и их подписи, а также заголовок в нашей форме (рисунок 4.7) так, как представлено на рисунке 4.11.

В заголовок формы введём другое название - «Инвентаризационная карта»и нажмем клавишу <Enter>(либо щелкнем мышью на свободном месте). С помощью буксировки мышью за верхнюю границу заголовка «Область данных» можно добиться нужного размера области для заголовка формы.

4.2.2 Вставка вычисляемого поля

Добавим в форму вычисляемое поле «Общая стоимость», равную произведению цены товара на его количество.

Выполним щелчок по кнопке **Поле**, а затем в нужном месте нашей формы. После размещения поле подписи будет содержать номер поля (в нашем случае, Поле 29), а содержимое поля – ссылку *Свободный*.

Первым делом двойным щелчком на поле наименования вызовем окно его свойств. Для свойства **Подпись** введем значение *Общая стоимость*. Остальные установки оставим без изменений и закроем окно свойств.

Затем заполним содержимое поля, открыв для этого двойным щелчком соответствующее окно свойств. В строку **Данные** одноименной вкладки поместим формулу вычисления содержимого. Любая формула представляет собой комбинацию из арифметических операторов и имен полей. Формула всегда должна начинаться со знака равенства, а имена полей в формуле необходимо заключать в квадратные скобки.

При создании вычисляемых полей необходимо учитывать следующее:

- ♦ имя поля в формуле должно совпадать с именем соответствующего поля в таблице, в противном случае появится сообщение об ошибке - #Имя?;
- ♦ ACCESS распознает в выражениях только те поля, которые были определены при создании таблицы и использованы при проектировании формы. Все остальные поля в вычислениях использованы быть не могут.

В нашем случае в строке *Данные* необходимо записать выражение

=[Цена]*[Кол-во]

и указать **Денежный формат**. Вид соответствующего окна свойств показан на рисунке 4.10.

Чтобы не заниматься вводом формулы с клавиатуры, можно воспользоваться кнопкой вызова построителя выражений , расположенной справа от строки **Данные**.

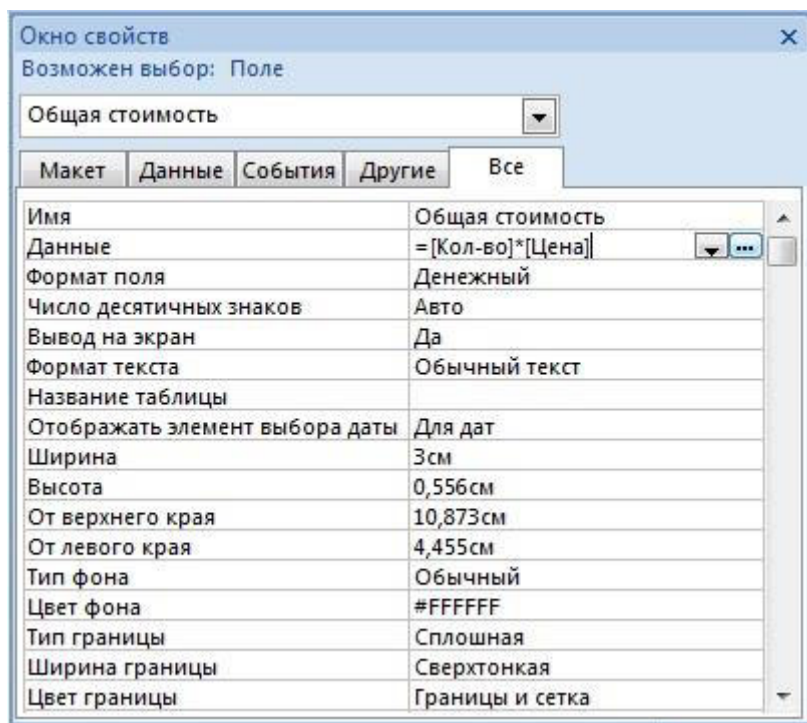


Рисунок 4.10 - Окно свойств поля «Общая стоимость»

4.2.3 Добавление надписи

Надпись – это управляемый элемент, содержащий текст, который вы хотели бы видеть в форме.

Создадим надпись «Удаление записи» в нижней правой части формы (рисунок 4.11). Для этого щелкнем сначала на кнопке **Надпись** панели инструментов, а затем в нужной области формы. В появившееся поле надписи введем текст «Удаление записи».

Форматирование текста надписи производится точно так же, как и в любом приложении MS Office.

Выполнив двойной щелчок на любом объекте формы, можно вызвать диалоговое окно **Свойства** соответствующего объекта, в котором задаются все его характеристики: размер, цвет фона, оформление и т.д. Таким образом можно, например, установить требуемый цвет и размер шрифта для заголовка формы, а также цвет фона для всей формы..

4.2.4 Создание флажков, списков, полей со списками

Если форма предназначена для ввода информации в БД, а не только для просмотра, то удобнее вводить данные в некоторые поля с помощью флажков, списков и полей со списками, так же, как это сделано в таблице «ТОВАРЫ» (поля «Поставщик», «Клиент», «Сертификат», «Ед-изм»). В этом случае не нужно, вводить с клавиатуры необходимые значения, а достаточно лишь мышью выбрать нужный вариант.

Обычно, если в таблице, на основе которой создавалась данная форма, поле уже было в виде флажка или поля со списком, то в получившейся форме это поле также будет отображаться флажком или полем со списком.

На примере поля «Сертификат» рассмотрим как его представить в виде флажка:

- 1 В режиме **Конструктор** удалим поле «Сертификат» из созданной формы, щелкнув на нем, чтобы выделить его, и нажмем клавишу <Delete>. Из формы будут удалены как само поле, так и его подпись;

- 2 Чтобы отобразить на экране список доступных для добавления полей, выполним последовательно: ☐ **Конструктор** ☐ **Сервис** ☐ **Добавить поля**

После этого в правой части окна Access появится окно **Список полей**;

- 3 Щелкнем на кнопке **Флажок** панели **Элементы управления**. Тем самым мы указываем тип поля, которое будет затем введено в форму;
- 4 Из списка полей буксировкой перетащим поле «Сертификат» на освободившееся после выполнения пункта 1 место. Как только будет отпущена клавиша мыши, ACCESS создаст флажок, связанный с полем «Сертификат».

Точно так же заменим текстовые поля «Поставщик», «Клиент», «Ед-изм» на поля со списками. В процессе их создания ACCESS запросит источник значений для этих полей, как это было описано ранее в пункте 2.5.

Получившуюся форму сохраним с именем «Инвентаризационная карта». Для этого выполним:

➤ **файл** ➤ **Сохранить**

Сохранение объекта... в: Инвентаризационная карта

☐ **ОК**

После выполнения всех описанных действий форма «Инвентаризационная карта» примет вид, показанный на рисунке 4.11.

Рисунок 4.11 Форма «Инвентаризационная карта» после добавления вычисляемого поля, надписи, а также создания полей со списками и флажка

4.2.5 Мастер кнопок

В форму можно добавить командную кнопку, которая будет при нажатии выполнять какое-либо действие, например, осуществлять заданные переходы по записям, производить заданную обработку записей и т.д. Для создания кнопки проще всего использовать **Мастер кнопок**, который начинает работать, если на панели элементов нажата кнопка **Использовать Мастера**.

Начало процесса создания кнопки – выполнение щелчка на элементе **Кнопка** в режиме конструктора, после чего на свободном месте формы нужно обрисовать мышью контур будущей кнопки. Затем начинает работать **Мастер** (открывается окно **Создание кнопок**).

Мастер содержит фиксированный набор кнопок и на его первом шаге надо выбрать подходящую. Названия кнопок, разбитые на категории, приведены в таблице 4.2.

В категории **Обработка записей** выберем кнопку **Удалить запись** и перейдем ко второму шагу **Создания кнопок**, где пользователь определяет изображение на кнопке. Это может быть произвольный текст или рисунок. **Мастер** предлагает набор стандартных рисунков, но разработчик может выбрать любой графический файл, нажав кнопку **Обзор**. Откроем рисунок под названием «Удалить запись».

Таблица 4.2 - Названия кнопок **Мастера** **Создание кнопок**

Категория «Переходы по записям»	Категория «Обработка записей»
Найти далее	Восстановить запись
Первая запись	Добавить запись
Поиск записи	Дублировать запись
Последняя запись	Печать записи
Предыдущая запись	Сохранить запись
Следующая запись	Удалить запись
Категория «Работа с формой»	Категория «Приложение»
Закрыть форму	Выход из приложения
Обновить данные формы	Запуск MS Excel
Открыть форму	Запуск MS Word
Печать текущей формы	Запуск блокнота
Применить фильтр формы	Запуск приложения
Категория «Работа с отчетом»	Категория «Разное»
Вывод отчета в файл	Запуск запроса
Отправка отчета	Запуск макроса
Печать отчета	Печать таблицы
Просмотр отчета	Набор номера

На третьем шаге **Создания кнопок** надо определить название кнопки (имя кнопки представляет собой переменную, по умолчанию не выводящуюся на экран) и его не надо путать *сподписью* – текстом, изображенным на кнопке. На этом работа с **Мастером** заканчивается.

Описание функции кнопки в форме можно выполнить с использованием элемента «надпись» панели элементов.

4.2.6 Проведение линий и рамок

Чтобы отделить группы полей, характеризующих какие-либо однотипные характеристики объекта в форме, можно с помощью элементов **Линия** и **Прямоугольник** изобразить соответствующие геометрические фигуры. Для этого нужно нажать нужную кнопку на панели элементов и с помощью мыши нарисовать линию или прямоугольную рамку.

Линии могут быть плоскими или рельефными. Для создания специальных эффектов после изображения (например, рамки) нужно щелкнуть на ней правой клавишей мыши и затем из появившегося контекстного меню выбрать пункт **Свойства**. В появившемся окне свойств линии необходимо выбрать пункт **оформление** и задать нужный эффект (Обычное, Приподнятое, Утопленное, Вдавленное, С тенью, Рельефное).

Переместим поля в нашей форме так, чтобы они создавали законченные функциональные группы, добавим элементы оформления, описанные в данном разделе, и в итоге получим окончательный вариант формы, показанный на рисунке 4.12.

Рисунок 4.12 - Окончательный вариант формы «Инвентаризационная карта»

4.3 Создание составной формы с перечнем

В тех случаях, когда между таблицами существует связь «один ко многим», возможно создание составной формы (формы с перечнем), в которой одной записи главной для этой формы таблицы соответствует одна или несколько записей подчиненной таблицы.

В качестве примера создадим составную форму для таблицы «ПОСТАВЩИКИ», в перечне которой содержатся сведения о всех товарах, относящихся к каждому конкретному поставщику.

Создание такой формы будем проводить с использованием **Мастера форм**, что значительно упрощает работу.

Выполним последовательность команд:

Создание форм Мастер форм

На экране появится диалоговое окно **Создание форм**, аналогичное показанному на рисунке 4.1, но с выбранной таблицей «ПОСТАВЩИКИ». Перенесем кнопкой **>>** все поля таблицы в окно **Выбранные поля**, затем в списке **Таблицы и запросы** активизируем таблицу «ТОВАРЫ» и перенесем в правое окно поля «Товар», «Ед_изм», «Кол-во», «Цена».

После нажатия кнопки **Далее** на экране появляется второе окно **Создание форм** (рисунок 4.13), в котором в качестве главной указываем таблицу «ПОСТАВЩИКИ» и выбираем вид **Подчиненные формы**. Дальнейшие шаги **Мастера форм** служат только для подтверждения установок по умолчанию и полностью аналогичны описанным в разделе 4.2. На последнем шаге зададим имя новой формы «Поставщики с перечнем».

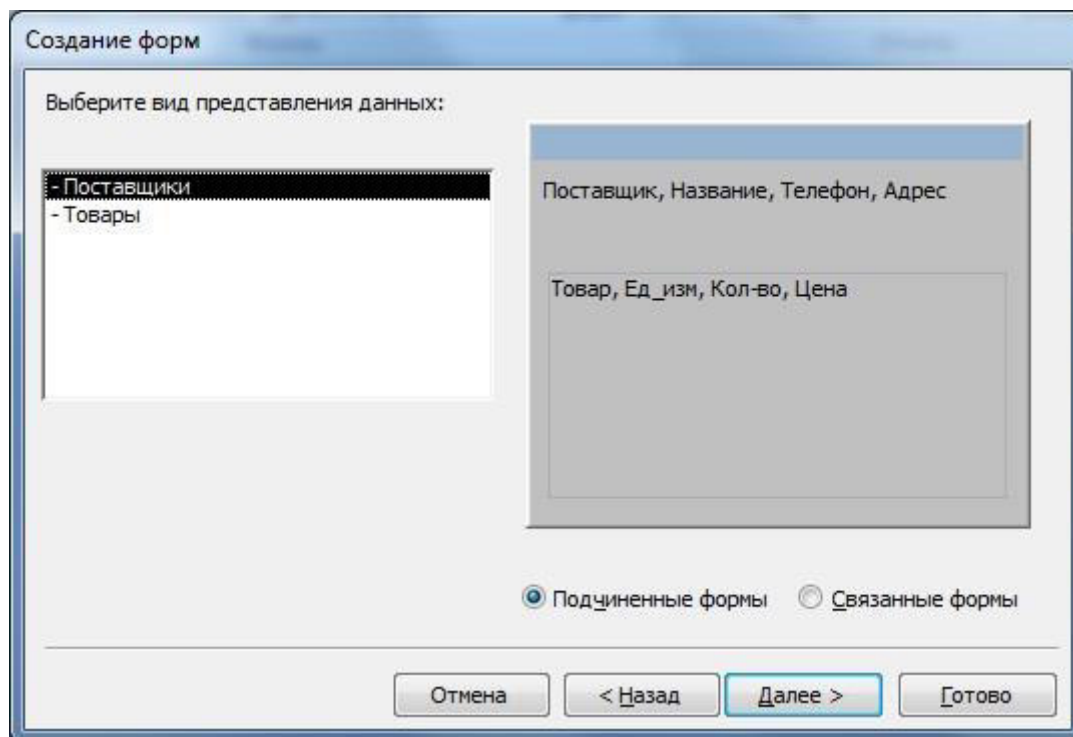


Рисунок 4.13 - Окно **Создание форм** при создании формы с перечнем

Фактически при рассмотренных действиях **Мастер форм** создает кроме основной еще и вспомогательную форму «**ТОВАРЫ подчиненная форма**», которая после завершения работы **Мастера форм** появляется в списке созданных форм базы данных.

Добавим в перечень вычисляемое поле «Сумма», равное общей стоимости каждой партии товара. Для этого откроем созданную форму «**Поставщики с перечнем**» в режиме конструктора. Окно, полученное при этом, имеет вид, показанный на рисунке 4.14.

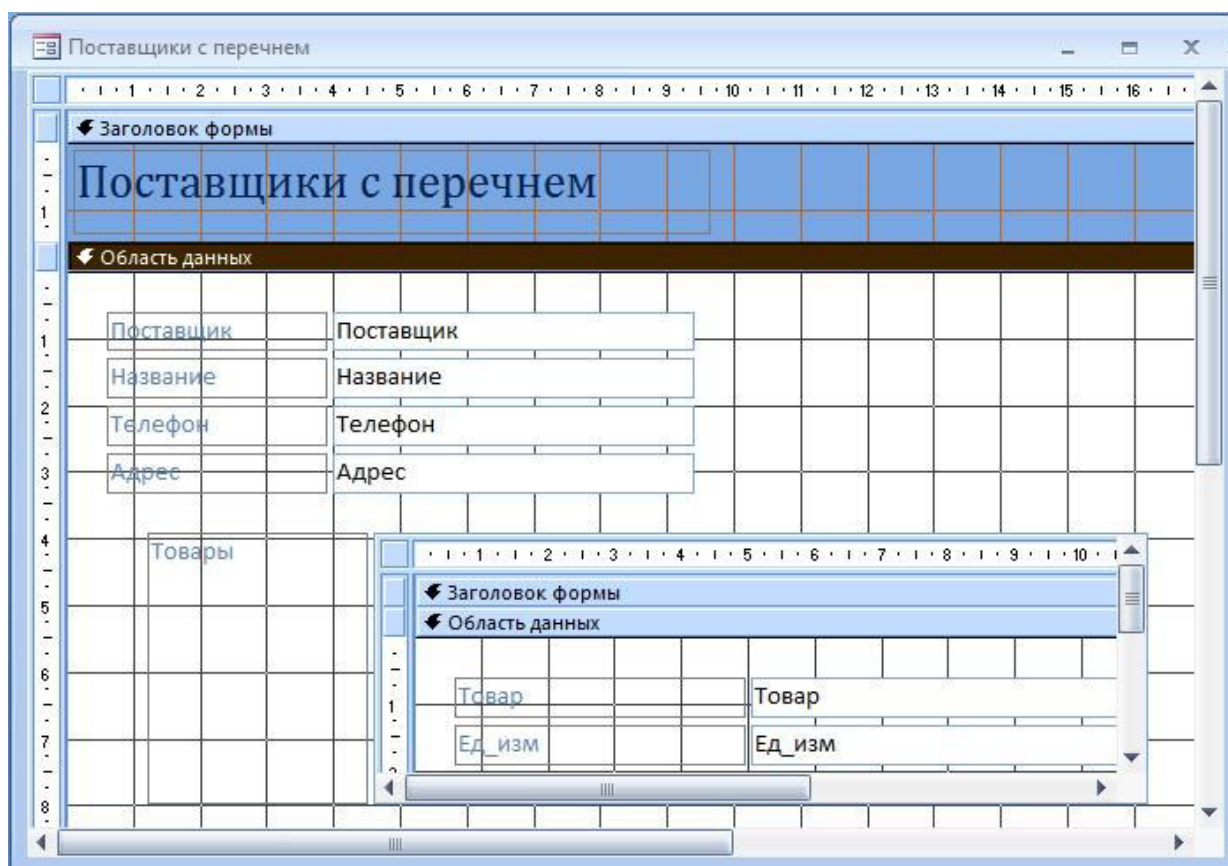


Рисунок 4.14 - Форма «**Поставщики с перечнем**» в режиме конструктора

Перечень в этом режиме может быть показан в виде пустого поля с соответствующим заголовком. Выполнив двойной щелчок мышью на этом поле, мы вызовем для редактирования подчиненную форму. Добавление в нее вычисляемого поля производится точно так же, как это описано в разделе 4.2.2. Вид формы после добавления поля «Сумма» показан на рисунке 4.15.

Рисунок 4.15 - Подчиненная форма с вычисляемым полем в режиме конструктора

Окончательный вид формы «Поставщики с перечнем» в режиме формы представлен на рисунке 4.16. Настроить ширину каждой колонки в таблице перечня можно непосредственно в режиме формы.

Товар	Ед_изм	Кол-во	Цена	Сумма
Красное бордо	бут	24	33,50р.	804,00р.
Сушёные улитки	шт	100	25,00р.	2 500,00р.
Икра из арбузов	кг	15	360,00р.	5 400,00р.
Лягушачьи лапки	кг	80	280,00р.	22 400,00р.
*		0	0,00р.	0,00р.

Рисунок 4.16 - Форма «Поставщики с перечнем»

5 ОТЧЕТЫ

Отчет создается, как правило, для вывода его на печать, его форматы соответствуют стандартным форматам бумаги.

Подобно запросам и формам, отчеты могут включать информацию из разных объектов БД (чаще всего из таблиц и запросов) и организовывать её по вашим указаниям.

Самый простой способ создать отчет – использовать инструмент **Мастер отчетов**, который будет рассмотрен ниже, а затем отредактировать его в режиме конструктора.

Создадим отчет, в котором содержатся сведения о всех товарах на складе, сгруппированные по покупателям. В отчете необходимо получить стоимость группы товаров для каждого покупателя и общую стоимость всех товаров на складе.

5.1 Мастер отчетов

Для запуска **Мастера отчетов** выполним следующую последовательность команд, после чего на экране появится окно **Создание отчетов** первого шага **Мастера отчетов**, показанное на рисунке 5.1.

☐ Создание ☐ Отчеты ☐ Мастер отчетов

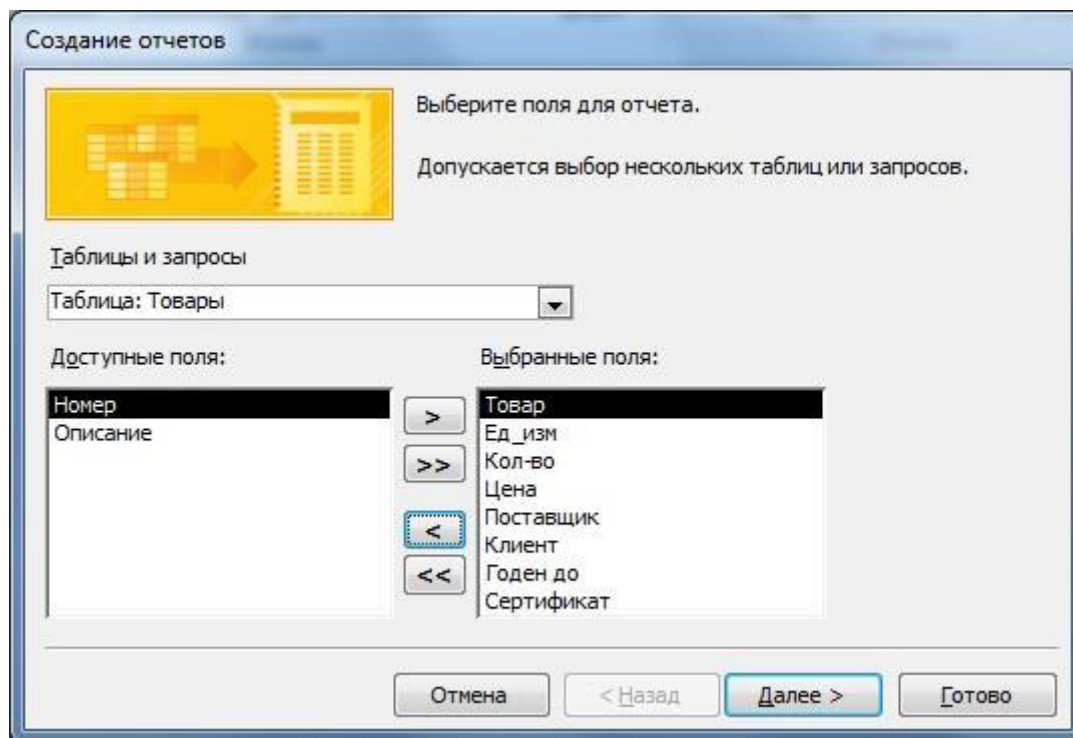


Рисунок 5.1 – Первый шаг **Мастера отчетов**

На этом этапе указываются исходная таблица или запрос и включаемые в отчет поля. Выберем все поля, кроме порядкового номера и описания, и перейдем в следующее окно **Создание отчетов**, показанное на рисунке 5.2.

Преимущество отчетов перед формами заключается в возможности группировки данных, что значительно улучшает внешний вид документа и повышает его читаемость. На втором шаге **Мастера отчетов** можно задать параметры группировки. Правая часть окна представляет собой упрощенный макет отчета, который наглядно показывает принцип группировки. Как правило, **Мастер отчетов** автоматически предлагает группировку по одному из полей. Поля, по которым данные будут сгруппированы, можно выбрать из

левого списка, для нашего отчета выберем поле «Клиент». Кнопка **Группировка** позволяет определить интервалы группировки и используется для задания диапазонов числовых данных и данных типа «дата/время», что позволяет избежать слишком большого числа групп.

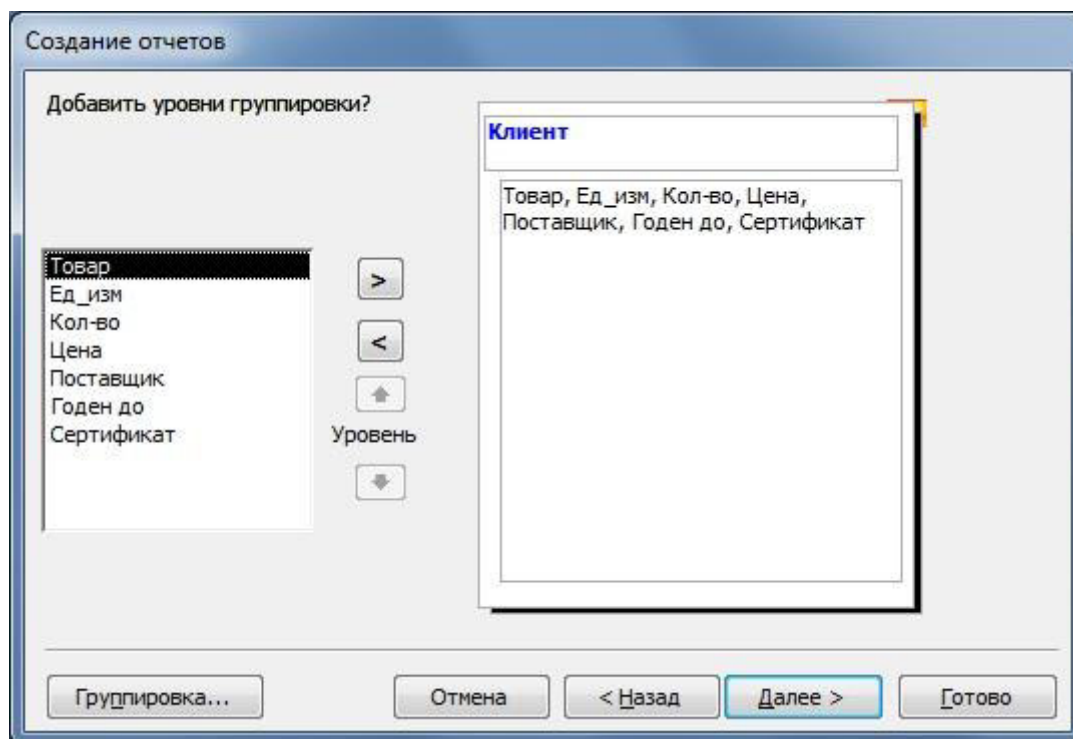


Рисунок 5.2 – Второй шаг **Мастера отчетов**

На третьем шаге (рисунок 5.3) выбирают поля, по которым будет выполнена сортировка данных.

ACCESS автоматически сортирует данные по тем полям, по которым выполняется группировка. Если записи должны быть отсортированы внутри группы, то поля сортировки указываются в этом окне.

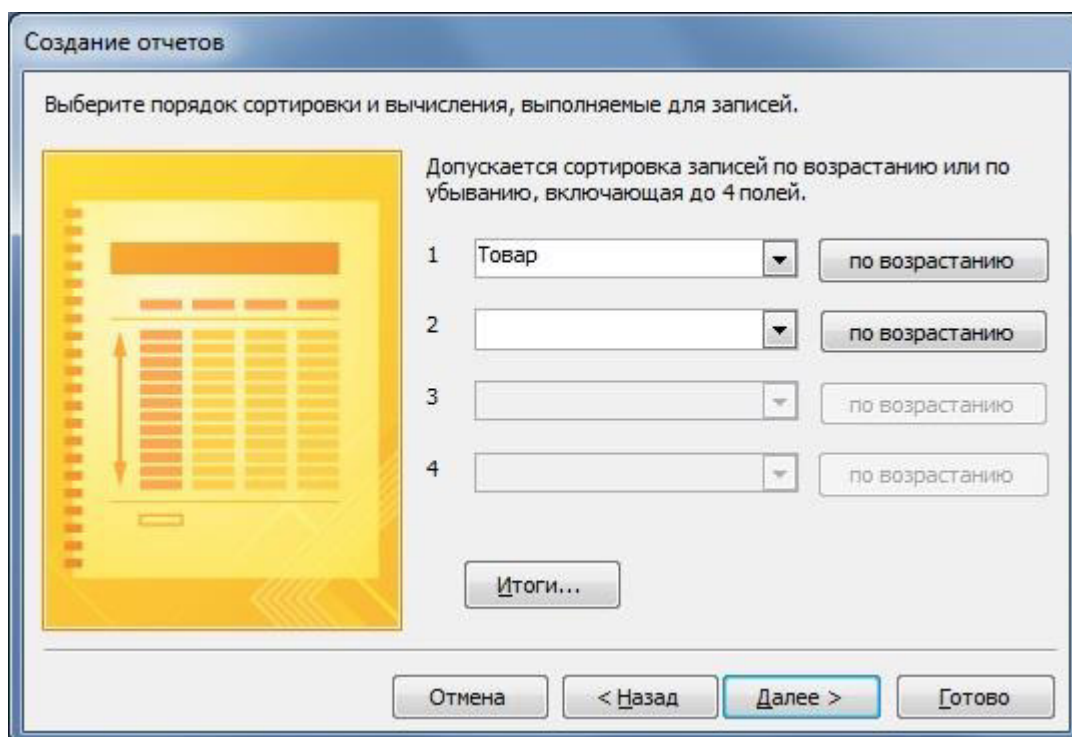


Рисунок 5.3 – Третий шаг **Мастера отчетов**

С помощью кнопки **Итоги** на этом же шаге можно задать вычисление некоторых итоговых значений для числовых полей. Диалоговое окно, получаемое при этом, показано на рисунке 5.4.

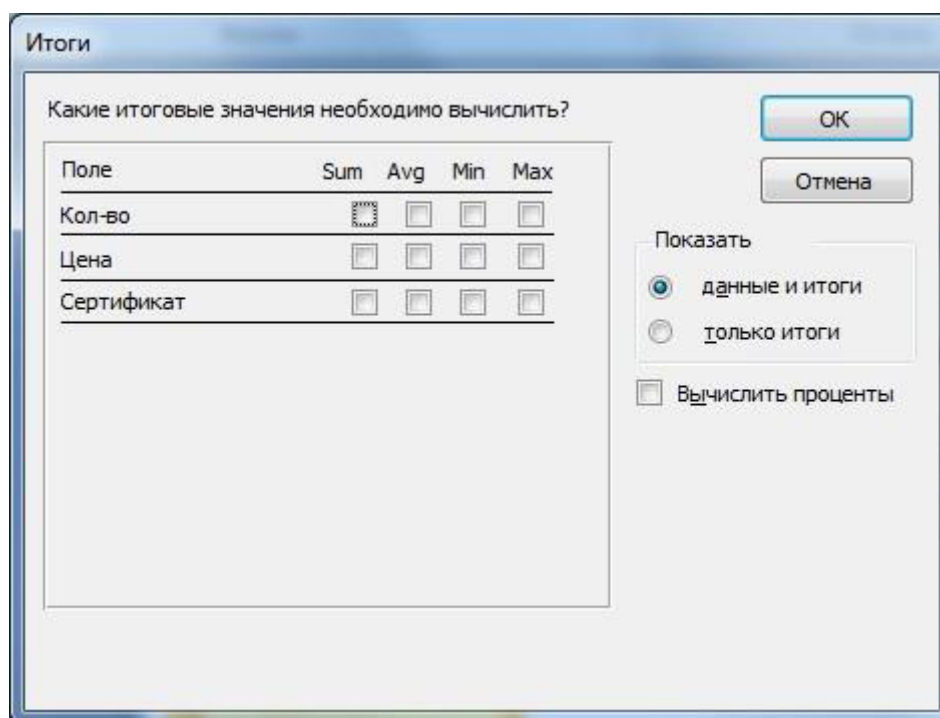


Рисунок 5.4 – Окно задания итогов отчета

На четвертом шаге (рисунок 5.5) задается общий вид отчета. Здесь вид окна **Мастера отчетов** зависит от того, как была задана операция группировки. В зависимости от этого программа предлагает набор макетов отчета. Картинка в левой части окна помогает понять, как будет выглядеть отчет при выборе соответствующего макета. Выберем, например, макет «ступенчатый».

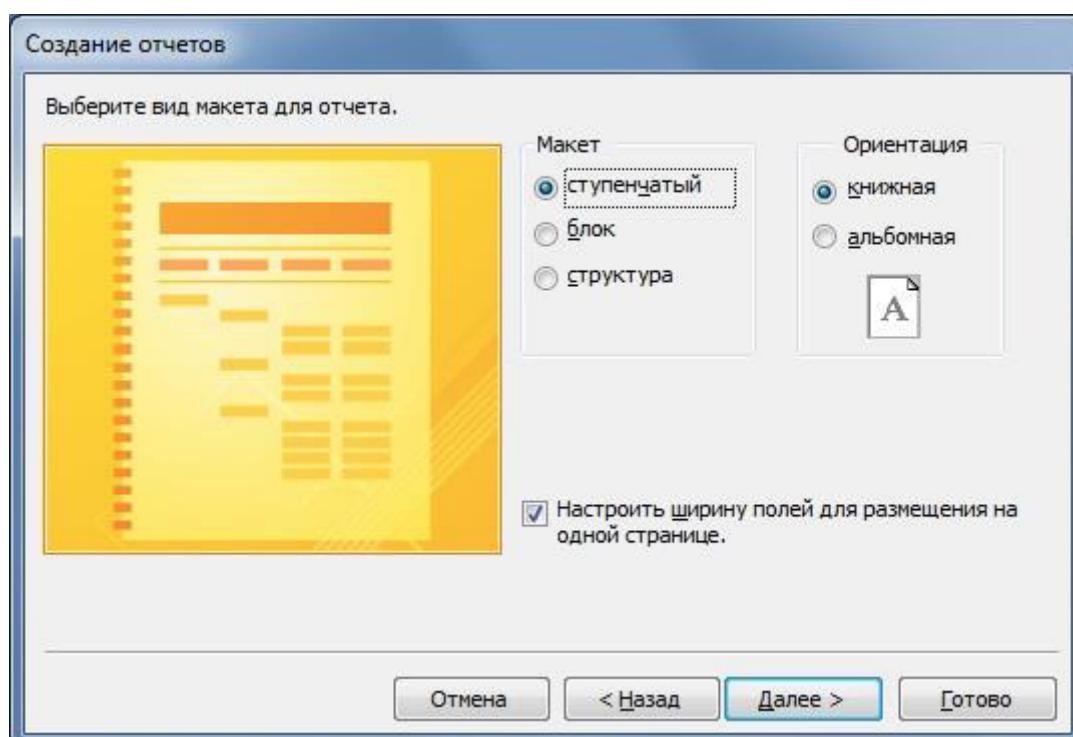


Рисунок 5.5 – Четвертый шаг **Мастера отчетов**

Пятый шаг (рисунок 5.6) позволяет определить стиль отчета, выберем «Стандартная». На шестом шаге просто задается имя отчета, оставим предлагаемое по умолчанию имя «ТОВАРЫ».

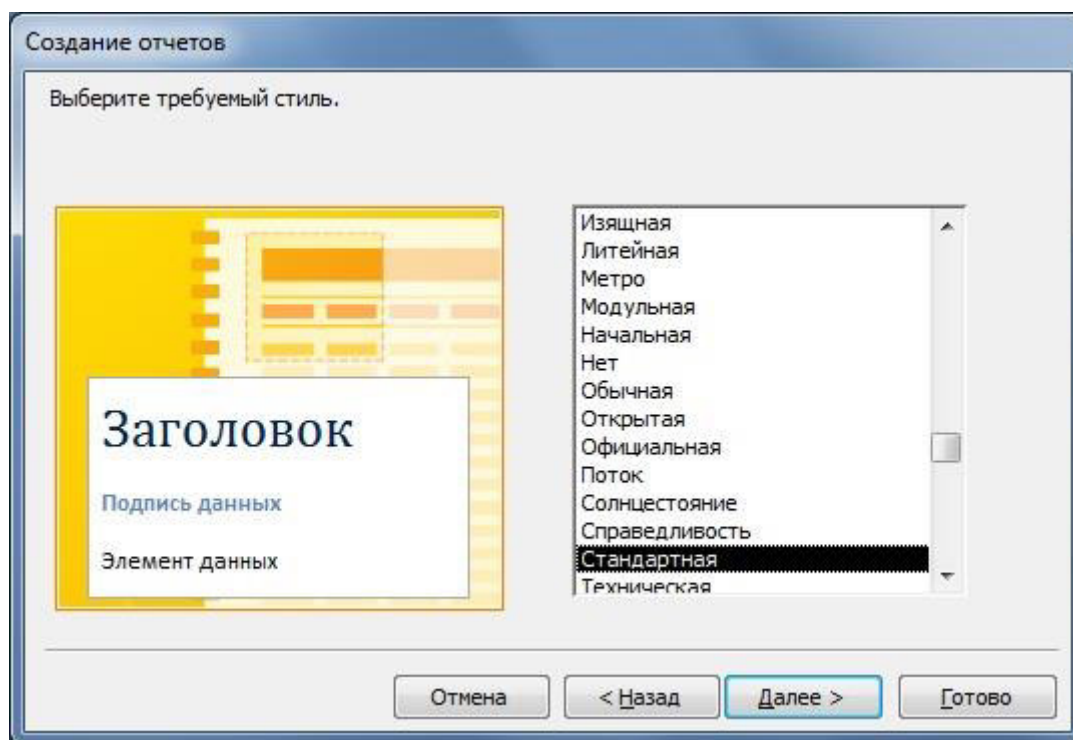


Рисунок 5.6 – Пятый шаг **Мастера отчетов**

Результат работы по созданию отчета с помощью **Мастера** показан на рисунке 5.7.

Товары

Клиент	Товар	Ед_изм	Кол-во	Цена	Поставщик	Годен до	Се
Абзац							
	Ананасы солён	шт	20	50,00р.	Fruct	31.12.2015	☐
	Анисовый сиро	л	30	64,20р.	Fruct	31.12.2015	☒
	Варенье из нер	л	60	78,60р.	Тибет	10.01.2020	☐
	Красное бордо	бут	24	33,50р.	Delikate	20.12.2018	☒
	Перепелиные я	шт	400	10,00р.	Восток	12.02.2016	☒
Амбра							
	Испанский сыр	кг	3	189,90р.	Восток	02.10.2017	☐
	Йогурт от лам	л	60	24,40р.	Тибет	10.10.2018	☒
	Кулебяка с фей	кг	60	99,90р.	Fruct	20.12.2015	☐
	Паста из бамии	кг	120	320,00р.	Selderey	20.06.2020	☐
Баран							
	Апельсиновый	бут	20	33,30р.	Fruct	26.04.2016	☐
	Заливное из су	кг	5	680,00р.	Tokyo	12.12.2016	☒
	Сушёные улитк	шт	100	25,00р.	Delikate	24.12.2017	☐

Страница: 1 | Нет фильтра

Рисунок 5.7 – Отчет, полученный при помощи **Мастера отчетов**

Отчет, показанный на рисунке, требует доработки. Если «ТОВАРЫ» вполне подходящее имя для объекта БД, то в заголовке документа должно быть что-то более традиционное и понятное. Надписи, определяющие заголовки колонок, расположены неправильно – некоторые буквы потеряны, для второй колонки не хватило места (названия товаров обрезаны).

Выходом из положения может стать уменьшение размеров шрифтов и позиционирование заголовков, что легко выполняется в режиме конструктора, как это было показано в предыдущем разделе.

5.2 Работа с отчетом в режиме конструктора

Созданный отчет «ТОВАРЫ» в режиме конструктора показан на рисунке 5.8.

Вся область отчета разделена несколькими горизонтальными линиями на зоны: **Заголовок отчета**, **Верхний колонтитул**, **Заголовок группы**, **Область данных**, **Примечание группы**, **Нижний колонтитул**, **Примечания отчета**.

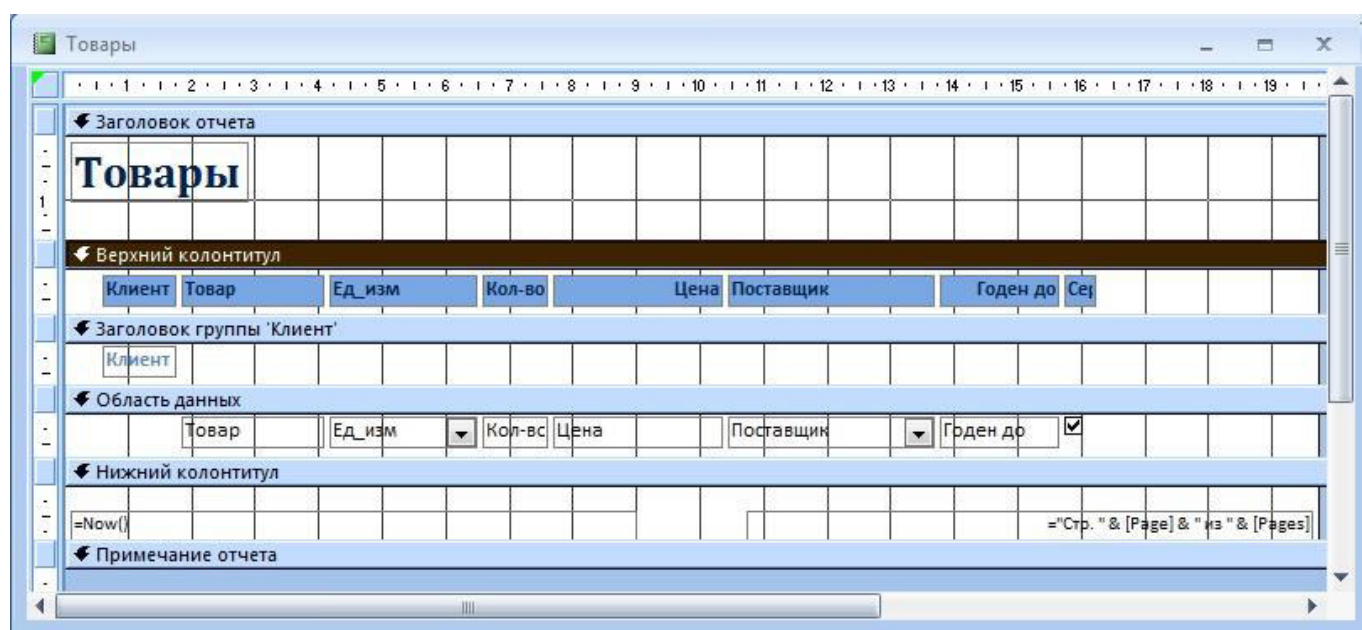


Рисунок 5.8 – Отчет «ТОВАРЫ» в режиме конструктора

Заголовок отчета расположен в верхней части первой страницы перед верхним колонтитулом. В области заголовка нашего отчета находится текст ТОВАРЫ.

Верхний колонтитул расположен в верхней части каждой страницы и обычно содержит заголовки столбцов таблицы.

Заголовок группы состоит из имени группы и слов *Заголовок группы*.

Область данных расположена между верхним и нижним колонтитулами и содержит записи выбранной таблицы или запроса.

Примечание группы появляется в конце группы записей и применяется, главным образом, для указания числа записей и/или суммы значений в полях группы. В нашем отчете примечание группы пока отсутствует.

Нижний колонтитул расположен в нижней части каждой страницы отчета и его используют для отображения номеров страниц и даты создания отчета.

Область применения отчета расположена в конце отчета и может содержать заключение по всему отчету.

Изменим заголовок отчета, для этого можно просто выделить его и ввести новый текст либо двойным щелчком мыши на заголовке вызвать диалоговое окно задания

свойств объекта и ввести новое наименование в строке **Подпись**. В этом же окне можно задать характеристики шрифта, его цвет и другие параметры заголовка.

Изменим позиционирование заголовков полей в верхнем колонтитуле и их размер, а также в соответствии с новым расположением заголовков, и распределение полей в области данных.

Для уменьшения расстояния между строками отчета просто уменьшим вертикальный размер области данных. Для этого установим указатель мыши на нижней ограничительной линии области (указатель примет при этом вид двунаправленной стрелки) и выполним буксировку вверх.

Вид отчета после выполнения указанных действий показан на рисунке 5.9.

The screenshot shows a report editor window titled "Товары". The report layout is as follows:

- Заголовок отчета:** Contains the main title "Товары для народа".
- Верхний колонтитул:** Contains a row of fields: "Клиент", "Товар", "Ед_изм", "Кол-во", "Цена", "Поставщик", "Годен до", and "Сертификат".
- Заголовок группы 'Клиент':** Contains the field "Клиент".
- Область данных:** Contains a table structure with columns: "Товар", "Ед_изм", "Кол-во", "Цена", "Поставщик", "Годен до", and a checkbox.
- Нижний колонтитул:** Contains a field "=Now()" and a page count formula "=Стр. " & [Page] & " из " & [Pages]".
- Примечание отчета:** Contains a field for a note.

Рисунок 5.9 – Модифицированный отчет «ТОВАРЫ» в режиме конструктора

5.3 Вычисления в отчетах

С помощью вычисляемых элементов управления обрабатываются отдельные записи, группы записей и все записи в отчетах.

Для отдельных записей вычисляемое поле нужно поместить в строку области данных отчета, для группы записей – в заголовок или в примечание группы, для всех записей – в заголовок или примечание отчета.

В процессе вычислений с использованием группы записей или всех записей применяются функции обработки ряда записей, приведенные в разделе 3.6 (суммирование, вычисление среднего и т.д.), и выражения, связанные с вычисляемым элементом.

Добавим в каждую строку отчета стоимость партии соответствующих товаров. Для этого, также как это было сделано при разработке формы, добавим в верхний колонтитул с помощью панели элементов надпись «Сумма», а в соответствующее место области данных вычисляемое поле $=[\text{Кол-во}] * [\text{Цена}]$.

Стоимость всей группы товаров для каждого покупателя разместим в области примечания группы, которая у нас пока отсутствует. Для создания этой области выполним команды:

☐ Конструктор ☐ Группировка

В появившемся в нижней части диалоговом окне **Группировка, сортировка и итоги**, сортировка и итоги в строке «Группировка **Клиент**» щелчком по кнопке **Больше** и выберем элемент списка **с разделом примечания** (рисунок 5.10).

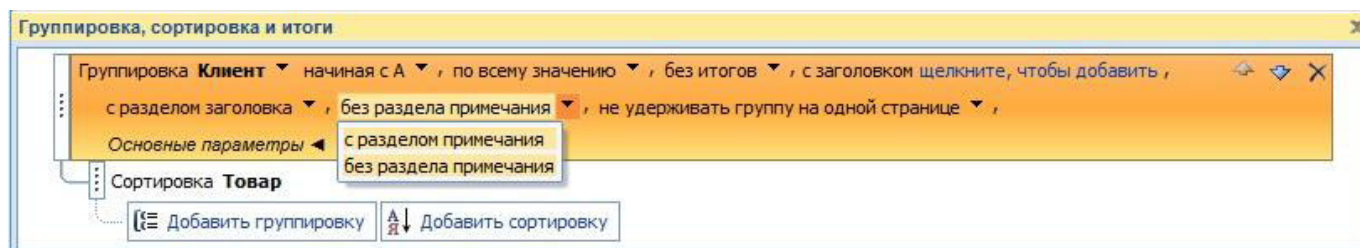


Рисунок 5.10 – Окно «Группировка, сортировка и итоги»

С помощью панели элементов введем в появившуюся в отчете область **Примечание группы «Клиент»** вычисляемое поле (также как это было описано в разделе 4.2.2). В поле наименований введем значение *Суммарная стоимость*, а в содержимом поля запишем формулу $\text{=Sum}([\text{Кол-во}]*[\text{Цена}])$.

Общую стоимость всех товаров на складе запишем точно так же в **Примечании отчета**, для чего добавим туда новое поле со значением наименования *Общая стоимость товаров на складе* и с содержимым $\text{=Sum}([\text{Кол-во}]*[\text{Цена}])$.

Для повышения удобочитаемости отчета внизу области **Примечаний группы** проведем горизонтальную линию, отделяющую одну группу от другой.

Окончательный вариант отчета в режиме конструктора показан на рисунке 5.11, а сам отчет – на рисунке 5.12.

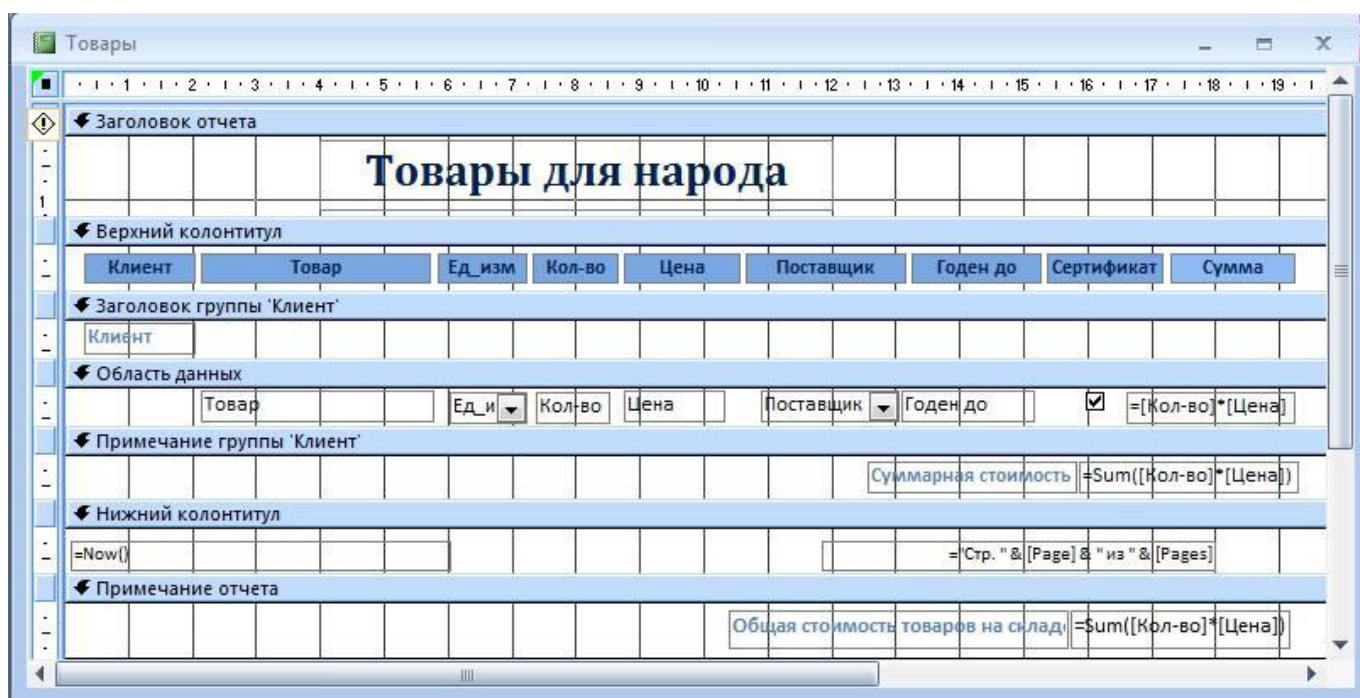


Рисунок 5.11 – Окончательный вариант отчета в режиме конструктора

Товары для народа

Клиент	Товар	Ед_изм	Кол-во	Цена	Поставщик	Годен до	Сертификат	Сумма
Абзац								
	Ананасы солёные	шт	20	50,00р.	Fruct	31.12.2015	☐	1 000,00р.
	Анисовый сироп	л	30	64,20р.	Fruct	31.12.2015	☒	1 926,00р.
	Варенье из нерпы	л	60	78,60р.	Тибет	10.01.2020	☐	4 716,00р.
	Красное бордо	бут	24	33,50р.	Delikate	20.12.2018	☒	804,00р.
	Перепелиные язычки	шт	400	10,00р.	Восток	12.02.2016	☒	4 000,00р.
Суммарная стоимость								12 446,00р.
Амбра								
	Испанский сыр	кг	3	189,90р.	Восток	02.10.2017	☐	569,70р.
	Йогурт от лам	л	60	24,40р.	Тибет	10.10.2018	☒	1 464,00р.
	Кул ебьяка с фейхуа	кг	60	99,90р.	Fruct	20.12.2015	☐	5 994,00р.
	Паста из бамии	кг	120	320,00р.	Selderey	20.06.2020	☐	38 400,00р.
Суммарная стоимость								46 427,70р.
Баран								
	Апельсиновый сок	бут	20	33,30р.	Fruct	26.04.2016	☐	666,00р.
	Заливное из суши	кг	5	680,00р.	Tokyo	12.12.2016	☒	3 400,00р.
	Сушёные улитки	шт	100	25,00р.	Delikate	24.12.2017	☐	2 500,00р.
Суммарная стоимость								6 566,00р.
Лебедь								
	Икра из арбузов	кг	15	360,00р.	Delikate	01.01.2017	☐	5 400,00р.
	Индийский чай	кг	50	124,00р.	Тибет	01.01.2019	☒	6 200,00р.
	Клюквенная подлива	кг	36	225,00р.	Selderey	10.12.2017	☒	8 100,00р.
	Солёные огурцы	шт	144	3,00р.	Fruct	16.06.2019	☒	432,00р.
	Тибетское пиво	бут	10	180,00р.	Тибет	01.01.2015	☒	1 800,00р.
Суммарная стоимость								21 932,00р.
Пиран								
	Лягушачьи лапки	кг	80	280,00р.	Delikate	30.01.2018	☒	22 400,00р.
	Слоновые хвосты	м	22	999,00р.	Тибет	10.10.2016	☒	21 978,00р.
	Сушёная говядина	кг	10	120,00р.	Восток	20.02.2017	☐	1 200,00р.
Суммарная стоимость								45 578,00р.
Общая стоимость товаров на складе								132 949,70р.

20 апреля 2016 г.

Стр. 1 из 1

Рисунок 5.12 – Отчет «ТОВАРЫ», открытый в режиме просмотра

6 ЗАДАНИЯ ДЛЯ ВЫПОЛНЕНИЯ НА ЛАБОРАТОРНЫХ РАБОТАХ

Данный раздел содержит задания для четырех лабораторных работ, выполняемых по ACCESS.

Первая работа «Создание баз данных» включает в себя этап проектирования, который выполняется дома, и этап создания и заполнения разработанной БД, выполняемый на компьютере.

Остальные работы проводятся с существующей базой данных, описание которой приведено в разделе 6.2.

6.1 Создание баз данных

На первом этапе работы требуется спроектировать и создать реляционную базу данных, состоящую из двух-трех связанных таблиц с типом связи «один ко многим». Тема БД ограничивается только фантазией автора и может простирается от описания домашней фонотеки до картотеки тайных операций ЦРУ. К данному этапу работы следует отнестись творчески: база данных должна иметь достаточное количество информации для решения поставленной задачи, но в то же время не быть слишком сложной. В таблицах необходимо использовать все типы полей, рассмотренных в данной работе. В одно из полей информация должна вводиться с помощью поля со списком (раздел 2.5 настоящего пособия). «Золотая середина» для количества полей: в основной таблице – $6 \div 8$, в связанных – $3 \div 4$.

В качестве возможных тем БД можно, например, использовать:

- ♦ «АВТОМАГАЗИН» – информация о покупателях, продажах и характеристиках автомобилей;
- ♦ «ТУРАГЕНТСТВО» – сведения о путевках, о приобретающих их лицах и о характеристиках туров;
- ♦ «АВТОНАРУШИТЕЛЬ» – данные о лицах, нарушивших правила дорожного движения, и характеристике нарушений;
- ♦ «БИБЛИОТЕКА» – сведения о книгах, имеющихся в библиотеке, и читателях;
- ♦ «ОЦЕНКА» – оценки, получаемые студентами на экзаменах по различным дисциплинам, и характеристики студентов (N° зачетки, дата сдачи экзамена, преподаватель, группа и т.д.).

По результатам проектирования должен быть составлен отчет, содержащий:

- 1) Общую характеристику спроектированной базы данных, ее назначение, область возможного использования.
- 2) Характеристики полей, включенных в таблицы БД. Их следует оформить так же, как это сделано в таблицах 1.1 – 1.3 или 6.1-6.3 настоящей работы. Для каждой таблицы следует пометить ключевое поле и поля для связи с другими таблицами. При необходимости дополнительно следует указать для некоторых полей наличие индекса, возможность повторения или уникальность данных, значения по умолчанию;
- 3) Структурную схему базы данных (аналогично рисунок 2.9 или 6.1);
- 4) Порядок заполнения таблиц. Здесь следует определить, какие поля должны быть заполнены раньше других, чтобы использовать их значения в качестве списка для другой таблицы. Если значения каких-то полей будут вводиться из отдельного списка, то этот список должен быть приведен.

Во время лабораторной работы спроектированная база данных должна быть реализована на компьютере (созданы и заполнены таблицы, организованы межтабличные связи).

Число записей в основной таблице должно быть не менее 10-ти, в дополнительных – не менее 3-х.

6.2 Описание базы данных NATALY

Во всех дальнейших заданиях используется готовая база данных «NATALY», расположенная в каталоге D:\WORK.



Перед началом работы с базой необходимо скопировать ее в свой каталог, переименовать (новое имя файла должно соответствовать вашей фамилии) и только потом загружать переименованную базу в ACCESS.

В базе данных «NATALY» хранится следующая информация. Имеется модельное агентство NATALY, названное в честь студентки группы 2АТ-18Д, предложившей данную структуру, в которой хранятся сведения о всех моделях (манекенщицах) города. Каждая модель служит в одном из трех бутиков (домов моды) и строго придерживается определенной диеты. Структура базы данных показана на рисунке 6.1, а характеристики полей – в таблицах 6.1 ÷ 6.3.

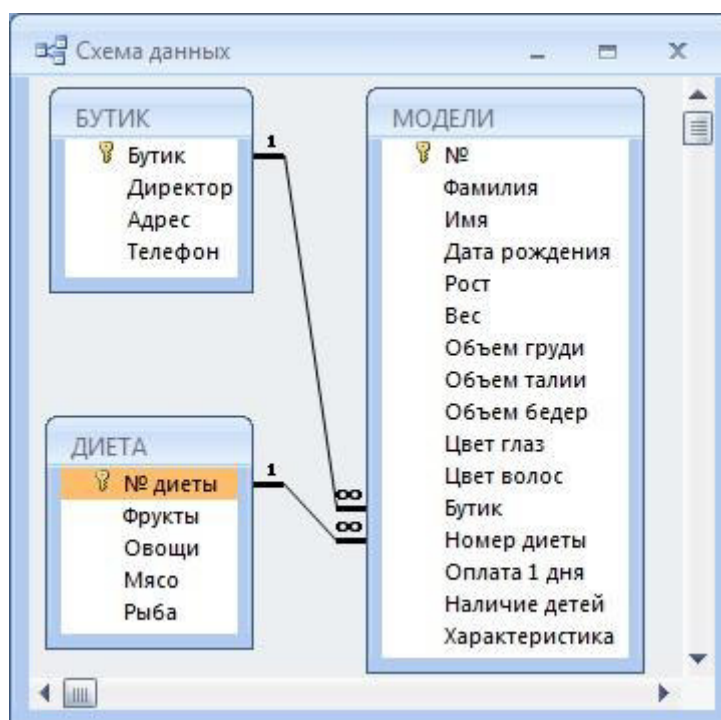


Рисунок 6.1 – Схема базы данных «NATALY»

Таблица 6.1 – Характеристика полей таблицы «ДИЕТА»

Имя поля	Тип поля	Длина поля	Примечание
№ диеты	текстовый	1	ключевое поле
Фрукты	числовой	Одинарное с плав.точкой	суточная норма фруктов [кг]
Овощи	числовой	Одинарное с плав.точкой	суточная норма овощей [кг]
Мясо	числовой	Одинарное с плав.точкой	суточная норма мяса [кг]
Рыба	числовой	Одинарное с плав.точкой	суточная норма рыбы [кг]

Таблица 6.2 - Характеристики полей таблицы «МОДЕЛИ»

Имя поля	Тип данных	Длина поля	Примечание
№	счетчик		ключевое поле
Фамилия	текстовый	13	
Имя	текстовый	8	
Дата рождения	дата		
Рост	числовой	Одинарное с плав.точкой	
Вес	числовой	Одинарное с плав.точкой	
Объем груди	числовой	целый, байт	
Объем талии	числовой	целый, байт	
Объем бедер	числовой	целый, байт	
Цвет глаз	текстовый	10	
Цвет волос	текстовый	10	
Бутик	текстовый	13	поле связи с таблицей «БУТИК»
Номер диеты	текстовый	1	поле связи с таблицей «ДИЕТА»
Оплата 1 дня	денежный		
Наличие детей	логический		
Характеристика	поле МЕМО		

Таблица 6.3 – Характеристика полей таблицы «БУТИК»

Имя поля	Тип поля	Длина поля	Примечание
Бутик	текстовый	13	ключевое поле
Директор	текстовый	17	
Адрес	текстовый	25	
Телефон	текстовый	7	

6.3 Запросы и фильтры

6.3.1 Создать запрос «БЛОНДИНКИ» на поиск моделей – блондинок с голубыми глазами. Вывести на экран фамилию, имя, дату рождения, цвет волос и глаз.

6.3.2 Создать запрос «МОЛОДАЯ МАТЬ» на поиск моделей, которые имеют детей и родились в 1980 году и позже. Вывести на экран фамилию, дату рождения и признак наличия детей.

6.3.3 Создать запрос «ИДЕАЛ» на поиск моделей, рост которых не меньше 170 см, объем груди – не меньше 90 см, талии – не больше 60 см, бедер – не меньше 90 см. Вывести на экран фамилию, имя, параметры поиска, бутик, где служат выбранные модели, и фамилию его директора.

6.3.4 Создать запрос «ОЧИ СИННИЕ» на поиск моделей, служащих в бутике «Луч», имеющих синий или голубой цвет глаз и рост не менее 170 см. Вывести на экран фамилию, бутик, его телефон, цвет глаз и рост выбранных моделей.

6.3.5 Создать запрос «Парижский луч» на поиск моделей из бутика «Парижский шик» по имени Лида и из бутика «Луч» с фамилией, начинающейся на букву М. Вывести на экран имя, фамилию и место службы модели.

6.3.6 Создать запрос «АМСТЕРДАМ» на поиск моделей, у которых имя начинается с буквы К, а фамилия с буквы М, либо вес удовлетворяет условию $60 \leq \text{Вес} \leq 65$. Вывести на экран имя, фамилию и вес выбранных моделей.

6.3.7 Создать запрос «SUPERSTAR» на поиск моделей из бутика «Луч», у которых отношение объема груди к объему талии ≥ 1.5 . Вывести на экран фамилию, объем талии, объем груди и вычисленное отношение.

6.3.8 С помощью запроса «КАТЯ» вычислить количество манекенщиц с таким именем и их средний рост.

6.3.9 Составить запрос «MINMAX» на вычисление количества моделей с диетой №3 из бутика, фамилия директора которого начинается на букву Ш, их минимального роста и максимального веса.

6.3.10 Однажды директор бутика «Луч» вспомнил, что обещал повысить на 100 руб. цену выступления одной из своих манекенщиц, но забыл кому именно. Помнил он только, что звали ее то ли Таня, то ли Катя, у нее были голубые глаза и не было детей. Составьте и выполните запрос на обновление «ПРЕМИЯ», который решает все проблемы директора.

6.3.11 Случилось так, что внезапно в бутике «Парижский шик» у всех брюнеток, придерживающихся диеты №2, и у шатенок с диетой №4 родились дети. Составьте и выполните соответствующий запрос на изменение «ДЕТИ».

6.3.12 Дирекция бутика «Луч» постановила уволить всех манекенщиц с детьми, родившихся раньше 1974 года. Подготовьте запрос на удаление «ДИСКРИМИНАЦИЯ», реализующий это несправедливое решение (выполнять запрос не нужно).

6.3.13 Во время очередных международных выступлений, где участвовали только модели, не имеющие детей с ростом ≥ 170 см, все блондинки с голубыми глазами и те шатенки с зелеными глазами, которые придерживались диеты №4, вышли замуж за миллионеров и уволились. Составьте соответствующий запрос «МЕЧТА» на удаление вышедших замуж моделей (выполнять запрос не нужно).

6.3.14 С помощью фильтра по выделенному найти все модели с именем «Катя».

6.3.15 С помощью фильтра по выделенному проверить результаты выполнения запросов 6.3.1, 6.3.5.

6.3.16 С помощью фильтра по выделенному найти фамилию модели из запроса 6.3.10.

6.3.17 С помощью обычного фильтра найти всех манекенщиц по имени «Катя» с голубыми глазами и «Наташа» с карими.

6.3.18 С помощью расширенного фильтра найти всех шатенок из бутика «Луч» с оплатой одного дня > 200 руб. и шатенок из бутика «Силуэт» с весом > 55 кг.

6.4 Формы

6.4.1 С помощью **Мастера форм** получить простую форму «Ф4-1» «в один столбец», содержащую все поля таблицы «МОДЕЛИ», за исключением порядкового номера.

6.4.2 С помощью **Конструктора** изменить размеры и положение полей в форме «Ф4-1», как это показано на рисунке 6.2. Сохранить отредактированную форму с именем «Ф4-2».

The screenshot shows a software window titled "МОДЕЛИ" with a form titled "Форма Ф4-2". The form contains the following fields:

Фамилия	Имя	Дата рождения	Бутик	Номер диеты
Иванова	Зина	23.03.1979	Парижский шик	3

Рост	Вес	Глаза	Волосы	Объем груди	Объем талии	Объем бедер
173,5	71	голубые	блондинка	90	60	90

Оплата 1 дня: 200р. Наличие детей: ☐

Характеристика:
Спортсменка, отличница, замечательно танцует, очень громко поет.

At the bottom, there is a status bar with the text "Запись: 1 из 40", "Нет фильтра", and a "Поиск" button.

Рисунок 6.2 – Форма «Ф4-2»

6.4.3 Ввести в форму «Ф4-2» вычисляемое поле, равное среднему значению объемов груди, талии и бедер моделей каждой из моделей. Сохранить полученную форму с именем «Ф4-3».

6.4.4 С помощью **Мастера форм** создать форму «Ф4-4», содержащую для каждой манекенщицы все поля таблицы «МОДЕЛИ», за исключением порядкового номера и характеристики, и все поля таблицы «ДИЕТА», за исключением поля «№ диеты».

6.4.5 С помощью **Конструктора** отредактировать форму «Ф4-4», придав ей вид, показанный на рисунке 6.3. Сохранить полученную форму с именем «Ф4-5».

Рисунок 6.3 – Форма «Ф4-5»

6.4.6 Ввести в форму «Ф4-5» вычисляемое поле «Стоимость питания», рассчитанное из условия, что один килограмм фруктов стоит 13 руб., мяса – 40 руб., рыбы – 50 руб., овощей – 10 руб. Сохранить полученную форму с именем «Ф4-6».

6.4.7 Изменить в форме «Ф4-6» вид поля «Наличие детей» на флажок, а поле «Бутик» - на поле со списком. Сохранить отредактированную форму с именем «Ф4-7».

6.4.8 Ввести в форму «Ф4-7» командную кнопку, выполняющую операцию закрытия формы. Сохранить результаты работы с именем «Ф4-8».

6.4.9 С помощью **Мастера форм** создать составную форму с перечнем, в перечне которой содержатся сведения о моделях, относящихся к каждому бутику. В основной форме должны отражаться все поля таблицы «БУТИК», а в перечне – фамилия, рост, вес и наличие детей. Сохранить полученную форму с именем «Ф4-9».

6.4.10 Ввести в перечень формы «Ф4-9» вычисляемое поле «Индекс Голда», равное разнице между ростом в см. и весом в кг. Отредактировать полученную форму, как это показано на рисунке 6.4 и сохранить с именем «Ф4-10».

Ф4-9

Форма Ф4-10

Бутик:

Директор:

Адрес:

Телефон:

СПИСОК МАНЕКЕНЩИЦ:

Фамилия	Рост	Вес	Наличие детей	Индекс Голда
Николаева	178,5	74,5	☐	104
Комарова	176,5	73	☐	103,5
Мамонтова	178	75	☒	103
Свешникова	169	64	☐	105
Нарышкина	173	69	☐	104
Барышева	168	64	☐	104
Рябова	162	59,5	☐	102,5
Лучева	169	66	☐	103

Записи: 1 из 13 | Нет фильтра | Поиск

Рисунок 6.4 – Форма Ф4-10

6.4.1 Создать форму «Ф4-11», добавив в форму «Ф4-9» вычисляемое поле, равное суммарному росту всех моделей соответствующего бутика.

6.5 Отчеты

6.5.1 С помощью Мастера отчетов создать табличный отчет без группировки, содержащий все поля таблицы «МОДЕЛИ», кроме «№» и «Характеристика», отсортированный по полю «Фамилия» в алфавитном порядке. Сохранить полученный отчет с именем «О5-1».

6.5.2 В режиме **Конструктора** отредактировать отчет «О5-1»:

- заменить заголовок на «Модели от NATALY»;
- расположить правильно заголовки колонок;
- переместить сами колонки в соответствии с заголовками;
- уменьшить расстояние между строками и/или размер шрифта так, чтобы отчет полностью помещался на одной странице.
- Сохранить отредактированный отчет с именем «О5-2».

6.5.3 В примечании отчета «О5-2» вычислить:

- суммарный вес всех моделей;
- средний рост манекенщиц;
- максимальный объем груди;
- минимальный объем талии.
- Сохранить результат в отчете с именем «О5-3».

6.5.4 С помощью **Мастера отчетов** создать отчет с группировкой по полю «Бутик» всех полей таблицы «МОДЕЛИ», кроме «№», «Цвет глаз», «Цвет волос», «Характеристика», отсортированный по дате рождения. Сохранить полученный отчет с именем «О5-4».

6.5.5 Добавить в каждую строку отчета «О5-4» вычисляемое поле «Оплата в месяц», вычисляемое в предположении, что каждая модель в месяц работает 10 дней. Отредактировать отчет, как это указано в задании 6.5.2 и сохранить результат работы с именем «О5-5».

6.5.6 Добавить в заголовок группы отчета «О5-5» вычисляемое поле «Всего в месяц», равное сумме полей «Оплата в месяц» для каждого бутика. В примечании группы разместить вычисляемые поля, заданные в задании 6.5.3. Сохранить отчет с именем «О5-6».

6.5.7 В примечании отчета «О5-6» разместить вычисляемое поле «Общая сумма в месяц», равное сумме полей «Оплата в месяц» для всех моделей и средние значения роста и веса всех манекенщиц. Сохранить отчет с именем «О5-7».

7 КОНТРОЛЬНЫЕ ВОПРОСЫ ДЛЯ ОЦЕНКИ УСВОЕНИЯ МАТЕРИАЛА

- 7.1 Данные каких типов могут храниться в полях БД? Приведите их характеристики.
- 7.2 Что такое плоские и реляционные таблицы ?
- 7.3 Какие виды связи между объектами вам известны ? Каким условиям должны удовлетворять поля для связи таблиц?
- 7.4 Что такое КЛЮЧ в базе данных?
- 7.5 Две таблицы <Магазин> и <Склад> базы данных имеют одинаковые по типу и размеру поля <Товар>. Ниже показано содержимое этих полей.

ключевое поле

соль

сахар

спички

неключевое поле

сахар

спички

соль

- а) Какого типа связь можно установить между таблицами по этим полям?
- б) Можно ли в какую-нибудь таблицу при наличии связи добавить запись со значением *мыло*? Если можно, то в какую?
- в) Можно ли в какую-нибудь таблицу при наличии связи добавить запись со значением *соль*? Если можно, то в какую?
- г) Можно ли из какой-нибудь таблицы при наличии связи удалить запись со значением *соль*? Если можно, то из какой?
- 7.6 На рисунке 7.1 представлено содержимое базы данных <Школа>. Запрос для вывода списка учеников 10 классов, 1988 года рождения, имеющих оценки не ниже 4, содержит выражение

	Фамилия	Год_рождения	Класс	Оценка
	Лыкова Ольга	1988	10	5
	Семенов Олег	1987	11	4
	Морозов Иван	1987	11	3
	Рыков Роман	1988	10	5
	Попов Сергей	1988	10	4
	Зайцева Марина	1987	10	5

Рисунок 7.1 Содержимое базы данных <Школа>.

- а) Класс=10 или Оценка>=5 или Год рождения=1988
- б) Оценка>=4 или Год рождения=1988 и Класс=10
- в) Класс=10 и Год рождения=1988 и Оценка=5 и Оценка=4
- г) Класс=10 или Оценка>=4 и Год рождения=1988
- д) Оценка>=4 и Год рождения=1988 и Класс=10
- 7.7 Задайте шаблон для поиска в базе данных <Школа> (рисунок 7.1) учеников из 11 класса, вторая буква фамилии которых есть О, а оценка 3 или 4
- 7.8 Какие записи будут найдены, если образец для поиска по полю <Фамилия> задан в виде шаблонов: а) Л[е-о]*а б) Л[е,о]*а ?
- | | | | |
|--------|---------|---------|---------|
| Левина | Лунина | Ленин | Лакина |
| Ликин | Ложкина | Люблина | Лионова |

7.9 Требуется найти записи о людях, родившихся раньше 1977 года. Выберите ВСЕ правильно заданные шаблоны поиска по полю <Дата рождения>, которое имеет тип Дата.

- а) <01.01.77 б) <=01.01.77 в) <01.01.78 г) <=01.01.78
 д) <01.77 е) <77 ж) <1977 з) <=31.12.76

7.10 На рисунке 7.2 показано окно создания запроса на выборку. Какие записи из таблицы <ТОВАРЫ> будут выбираться? Как изменятся результаты запроса, если условие (20 Or 50) переместить на одну строку выше?

Поле:	Товар	Цена	Годен до	Кол-во
Имя таблицы:	ТОВАРЫ	ТОВАРЫ	ТОВАРЫ	ТОВАРЫ
Сортировка:				
Вывод на экран:	☒	☒	☒	☒
Условие отбора:		>100	<#01.01.2006#	20 Or 50
или:				

Рисунок 7.2 Окно создания запроса на выборку

7.11 На рисунке 7.3 показано окно запроса на выборку с использованием групповых операций для базы данных, показанной на рисунке 7.1. Каков будет вид результата выполнения данного запроса?

Поле:	фамилия	Год_рождения	Класс	Оценка
Имя таблицы:	Школа	Школа	Школа	Школа
Групповая операция:	Min	Count	Sum	Группировк
Сортировка:				
Вывод на экран:	☒	☒	☒	☒
Условие отбора:				3 Or 4
или:				

Рисунок 7.3 Окно создания запроса на выборку с групповыми операциями

ЛИТЕРАТУРА

1. ACCESS XP для занятых.-СПБ:Питер,2005.- 312 с.:ил.
2. Гончаров А. ACCESS XP в примерах.-СПБ:Питер,2003.-386 с.:ил.
3. Золотарюк А. Технология работы с MicrosoftOffice.: [учеб. пособие].- М.:Академический Проект, 2002.-416 с.:ил.
4. Кузнецов А. MicrosoftAccess 2003: Русская версия: Учебный курс.- СПб:Питер,2005.-363 с.:ил.
5. Карпов Б. MicrosoftAccess 2000: Справочник. СПб.: Питер, 2000. 416 с.
6. Михеева В., Харитонов И. MicrosoftAccess 2003.-БХВ-Петербург,2004. -386 с.:ил
7. Николаев Н.А. Система управления базами данных MicrosoftAccess 2003. Методическое пособие по курсу “Информатика”. Новоуральск, НГТИ, 2005. – 68 с.: ил.
8. Робинсон С. MicrosoftAccess 2000: Учеб. курс/ Пер. с англ. А.Зеленина. СПб.: Питер, 2002. -512 с.
9. Сеннов А. Access 2003. Практическая разработка баз данных:[Учебный курс]. - СПб:Питер,2004.-256 с.:ил.
10. Советов Б. Я. Базы данных : теория и практика: учеб. для бакалавров. - М. :Юрайт, 2013. – 399 с. : ил.
11. Фуфаев Э. В. Базы данных: учеб. пособие.- М. : Академия, 2013. – 320 с. : ил.

Электронные учебники (каталог Z:\Education\MSOffice сервера кафедры ИиП):

1. MicrosoftAccess 2000. Шаг за шагом: Практ. пособ. / Пер. с англ. М.: Издательство ЭКОМ, 2002. — 352 с.: илл.
2. Андерсен В. Базы данных MicrosoftAccess. Проблемы решения: Практ. пособ. / Пер. с англ.—М.: Издательство ЭКОМ, 2001.—384 с.: илл.
3. Иллюстрированный самоучитель по Access 2002

Автор:

Орлова Ирина Викторовна

СИСТЕМА УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ MICROSOFT ACCESS 2010

Методическое пособие по курсу «Информатика», «Информационные технологии»
для студентов всех специальностей очной формы обучения.
Новоуральск, НТИ НИЯУ МИФИ, 2016, 68 с.

Сдано в печать

Печать плоская

Заказ Издательство НГТИ

Формат А5

Уч.-изд.л. 2.0

Лицензия ИД №00751

Бумага писчая

Тираж 50 экз.

г.Новоуральск, Ленина, 85