

Министерство образования и науки Российской Федерации

Владивостокский государственный университет
экономики и сервиса

БАЗЫ ДАННЫХ

Практикум

Владивосток
Издательство ВГУЭС
2005

Практикум составлен в соответствии с требованиями государственного стандарта России и содержит учебно-методические указания в виде контрольных примеров в справочную информацию по теоретическому материалу, необходимых для выполнения лабораторных работ, а также контрольные вопросы для самостоятельной проверки освоенного материала.

Предназначен по дисциплине «Базы данных» для студентов специальности 351400 «Прикладная информатика в экономике», изучающих ее в качестве дисциплины общепрофессионального цикла и специальности 061800 «Математические методы в экономике», изучающих ее в качестве дисциплины по выбору в цикле специальных дисциплин.

ББК 32.973/2-018

Составители: Бедрина С.Л., ст. преп. кафедры ИИКГ
Богданова О.Б., ст. преп. кафедры ИИКГ
Маевская М.Л., ст. преп. кафедры ИИКГ

Печатается по решению РИСО ВГУЭС

© Издательство Владивостокского
государственного университета
экономики и сервиса, 2005

ВВЕДЕНИЕ

Издание представляет собой доступное практическое пособие для работы с системой управления базами данных MS Access для Windows. В нем содержатся сведения о технологиях создания таблиц базы данных и схемы взаимосвязей между ними. Дается методика разработки и использования форм для ввода, корректировки и просмотра, применения диалоговых средств выборки и обработки данных, конструирования отчетов в виде печатных документов. Также уделяется внимание формированию макросов и программированию модулей на языке Visual Basic for Applications, использованию диспетчера кнопочных форм для создания приложений пользователя с удобным графическим интерфейсом, работе с базой данных в многопользовательском режиме.

Рассматривается разработка задач и создание целостного приложения для отдельно взятой предметной области. Учитывая простоту создания мастером баз данных типовой базы данных в предлагаемых разработчиками областях, основное внимание уделяется другой технологии – технологии построения оригинальной базы данных, спроектированной пользователем. Эта технология позволяет проникнуть глубже в суть принципов построения и работы реляционной базы данных (РБД) и лучше освоить инструментальные средства Access. При этом шаг за шагом прослеживаются операции по созданию БД, форм, запросов и отчетов.

Рассматриваемая технология использования разнообразных средств СУБД Access позволит успешно применять данный пакет для решения практических задач.

MS Access – это система управления базами данных (СУБД). Как и другие продукты этой категории, Access предназначен для хранения и получения данных, представления их в удобном виде и автоматизации часто выполняемых операций (например, для ведения счетов, учета материальных ценностей, планирования и т.п.).

С другой стороны, Access является мощным приложением Windows. Система Access работает под управлением Windows; таким образом, все преимущества Windows доступны в Access. В то же время Access – это реляционная СУБД. Это означает, что с помощью Access можно получить доступ к любым данным любого типа и использовать одновременно несколько таблиц базы данных. Использование реляционной СУБД позволяет упростить структуру данных и, таким образом, облегчить выполнение работы.

1. ОРГАНИЗАЦИОННО-МЕТОДИЧЕСКИЕ УКАЗАНИЯ

1.1. Цель и задачи лабораторного практикума

Целью выполнения лабораторных работ является получение практических навыков по разработке реляционной базы данных, созданию таблиц баз данных и схем взаимосвязей между ними, а также создания приложений пользователей для обработки информации из базы данных на примере конкретной предметной области.

В ходе выполнения лабораторных работ у студента должно формироваться представление о принципах и способах обработки информации с использованием технологии баз данных.

В ходе достижения цели решаются следующие задачи:

- развитие логического и алгоритмического мышления;
- изучение принципов работы прикладного программного обеспечения;
- выработка умения самостоятельного решения задач по анализу информации и выбору метода ее обработки;
- изучение технологии работы с базами данных;
- изучение возможностей языков описания данных и манипулирования данными;
- получение навыков в разработке приложений пользователя и их тестирования;
- изучение перспектив развития технологий создания баз данных.

1.2. Требования к знаниям и умениям, приобретаемым при выполнении лабораторного практикума

В результате выполнения лабораторных работ студент должен знать:

- основные методы анализа информационных процессов;
- информационные закономерности, специфику информационных объектов и ресурсов, информационных потребностей в предметной области;
- основные принципы организации БД информационных систем, способы построения БД;
- способы представления текстовой и нетекстовой информации в ИС;
- возможности новых информационных технологий;
- средства реализации алгоритма на языке программирования в выбранной СУБД (например, Visual Basic for Application (VBA)).

Кроме того, студент должен уметь:

- применять инструментальные средства, поддерживающие разработку программного обеспечения профессионально ориентированных ИС;
- применять современные системные программные средства управления БД;
- формулировать запросы на языке SQL;
- представлять результаты решения отдельных задач в удобном для восприятия виде.

1.3. Объем и сроки изучения курса

Курс «Базы данных» для специальности 351400 «Прикладная информатика в экономике» общим объемом 122 часа изучается в течении 5 семестра. Курс «Базы данных и знаний» для специальности 061800 «Математические методы в экономике» общим объемом 119 часа изучается в течение 6 семестра

1.4. Техническое и программное обеспечение для выполнения лабораторного практикума

Для выполнения лабораторных работ необходимо наличие персонального компьютера не менее Pentium III-500МГц с оперативной памятью не менее 96 Мбайт и памятью на жестком диске 8 Гбайт и выше. На персональных компьютерах должно быть установлено следующее программное обеспечение: операционная система Windows 2000 и выше, а также интегрированный пакет MS Office 2000 и выше.

1.5. Виды контроля знаний студентов и их отчетности по лабораторному практикуму

Каждая выполненная лабораторная работа должна быть проверена преподавателем и защищена студентом с учетом высказанных замечаний. По итогам защиты студент получает определенное количество баллов. Допуск к семестровой аттестации по дисциплине «Базы данных» осуществляется только при выполнении всех лабораторных работ.

2. ЛАБОРАТОРНЫЙ ПРАКТИКУМ ПО КУРСУ БД

Общие указания к выполнению лабораторных работ

Целью выполнения лабораторного практикума является приобретение навыков работы с базами данных и приложениями для работы с БД (СУБД Access).

В качестве предметной области для лабораторных работ выбрана некоторая коммерческая фирма, которая выполняет заказы на поставку какого-либо вида товара. Клиентами фирмы являются юридические лица (фирмы). Список товаров реализуемых фирмой не ограничен.

В качестве входной информации для разрабатываемой системы выступает информация о клиентах (наименование, реквизиты, предоставленные скидки), информация о товаре (наименование, цена и др. согласно варианту), а также данные о произведенных заказах. Следует учитывать, что заказ, выполняемый фирмой, может включать любые наименования товара в любом количестве.

В качестве выходной информации могут быть представлены: прайс-лист, отчеты по продажам различных товаров, отчеты по фирмам и т.д.

В разрабатываемой системе должны быть реализованы следующие функции:

- Ввод и редактирование данных
- Поиск и выборка данных по различным критериям, имея в виду вероятные запросы пользователей
- Выдачу выходных документов

ЛАБОРАТОРНАЯ РАБОТА № 1

Создание таблиц и связей между ними

Цель работы. Построение инфологической модели предметной области, определение состава базы данных, выбор модели данных и СУБД, разработка базы данных: таблиц для хранения информации, связей между ними, ограничений на значения полей.

Контрольный пример

Реализация общей задачи разбита на несколько этапов:

1. В результате анализа поставленной задачи составим инфологическую модель предметной области. Инфологическая модель (Рис. 1) описывается тремя сущностями (**Фирма**, **Товар**, **Заказ**), атрибутами (обязательными), характеризующими каждую сущность, и связями между сущностями (фирма-заказ, товар-заказ).

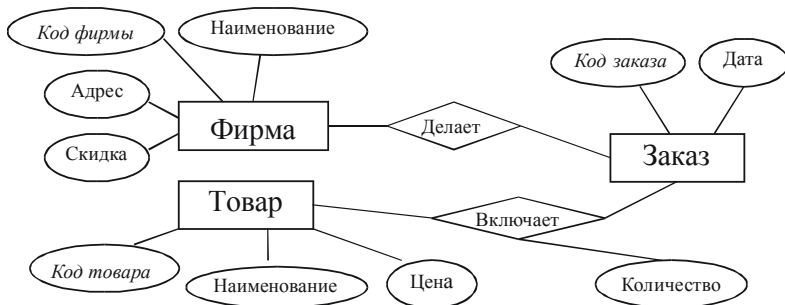


Рис. 1. Инфологическая модель предметной области

2. На основании этой модели, с учетом того, что конкретная реализация будет выполнена в среде СУБД Access, разрабатываем схему данных для каждой будущей таблицы (базовый вариант):

Таблица **Фирма**

Наименование полей	Тип	Примечание
Код_фирмы	Счетчик	Ключевое поле
Наименование_ф	Текстовый	
Адрес	Текстовый	
Скидка	Числовой	

Таблица **Товар**

Наименование полей	Тип	Примечание
Код_товара	Счетчик	Ключевое поле
Наименование_т	Текстовый	
Цена	Числовой	

Таблица **Заказ_Фирма**

Наименование полей	Тип	Примечание
Код_заказа	Счетчик	Ключевое поле
Код_фирмы	Числовой	
Дата	Дата/время	

Таблица **Заказ_Товар**

Наименование полей	Тип	Примечание
Код_заказа	Числовой	Ключевое поле
Код_товара	Числовой	Ключевое поле
Количество	Числовой	

3. Создание таблиц в СУБД Access.

Для этого необходимо выбрать закладку *Таблицы* (пункт списка «Объекты») и нажать кнопку *Создать* и выбрать режим *Конструктор* или изначально выбрать пункт *Создание таблицы в режиме конструктора* рабочей области окна, после чего описать первую таблицу **Фирма** (описываются заголовки столбцов, указываются типы полей и ключевое поле). (Таблицы типа **Фирма** принято называть справочниками).

Тип поля выбирается из списка.

Для поля **Скидка** установим размер поля – *Одинарное с плавающей точкой* и формат поля – *Процентный*.

Для создания ключевого поля необходимо вызвать контекстное меню для нужной строки и выбрать пункт «Ключевое поле» или воспользоваться пиктограммой панели инструментов.

Перед закрытием таблицу необходимо сохранить и дать ей соответствующее имя.

Результат представлен на рис. 2.

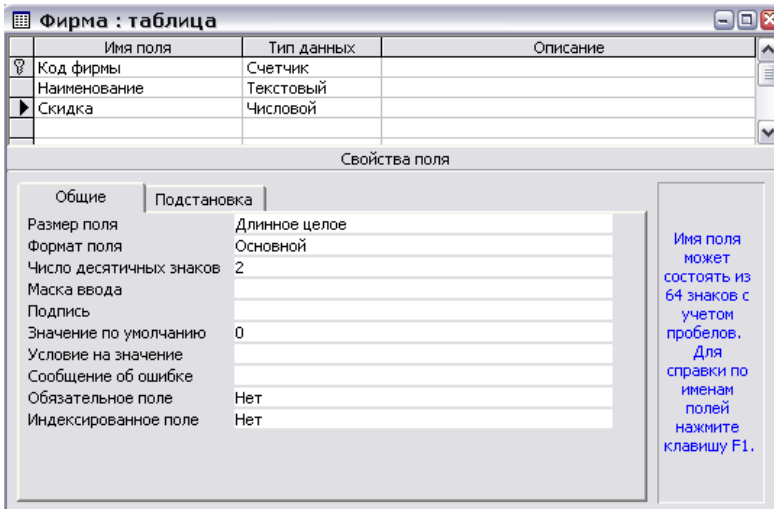


Рис. 2. Описание структуры таблицы Фирмы

Аналогичным образом создаем остальные таблицы. Две последние таблицы **Заказ_Фирма** и **Заказ_Товар** представляют собой описание связей между объектами и помимо уникальных данных, которые описываются обычным способом, содержат поля из таблиц справочников. Следовательно, при описании полей таблицы в режиме Конструктора можно воспользоваться закладкой *Подстановка* и выполнить следующие настройки:

Тип элемента управления – Выбрать *Поле со списком*

Тип источника строк – Выбрать *Таблица или запрос*

Источник строк – указывается таблица, которая является источником данных.

Например, для поля **Код_Фирмы** из таблицы **Заказ_Фирма**, источником является таблица **Фирма**

Указать источник данных для полей:

Код_Фирмы – Таблица **Заказ_Фирма** (источник – коды фирм из таблицы **Фирма**)

Код_Товара – Таблица **Заказ_Товар** (источник – коды товаров из таблицы **Товар**).

В таблицах заказов отображаются и содержатся коды таблиц справочников. Для того чтобы отображались значения при подстановке указать число столбцов – 2, в пункте *Ширина столбцов* указать 0 см; 4 см.

Пример такого описания представлен на рис. 3.

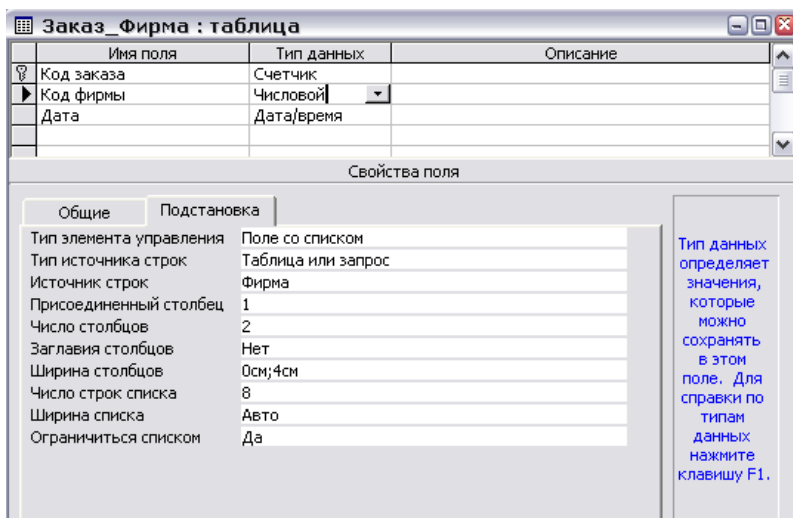


Рис. 3. Описание полей с подстановкой

4. Создание схемы данных. Для того, чтобы описать связи между таблицами, используем пункт меню Access *Сервис~Схема данных*.

Связывание полей проведем способом перетаскивания поля из одной таблицы в другую. Например, поле из таблицы **Фирма** *Код_фирмы* необходимо переместить в таблицу **Заказ_Фирма** на аналогичное поле. *Следует отметить, что поля, по которым выполняется связь, должны быть одинаково описаны* (допускается связь полей с типами Счетчик и Числовой). В окне установки связей выполняются настройки рис. 4.

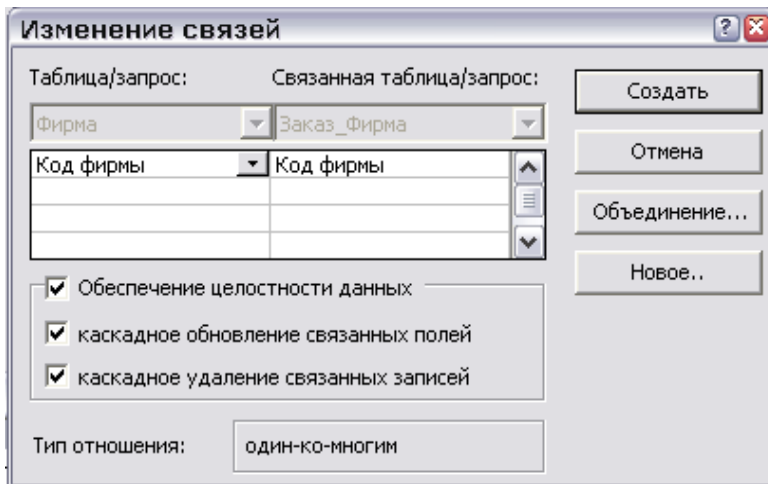


Рис. 4. Установка связей между таблицами

Результат установления связей в базе данных представлен на рис. 5.

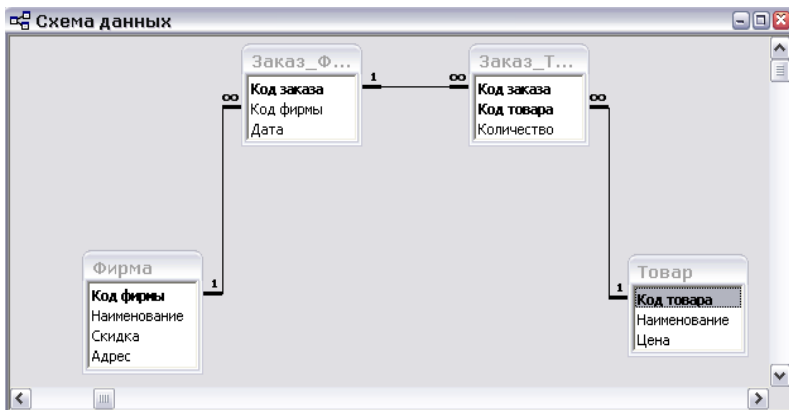


Рис. 5. Схема данных

Индивидуальное задание

1) Определить список необходимых таблиц и дополнительных атрибутов, согласно своему варианту по прилагаемой ниже таблице.

2) Создать дополнительную таблицу или добавить новые атрибуты в модель. Если необходимо, откорректировать схему данных («привязать» новую таблицу).

3) Заполнить таблицы данными: не менее 5 строк для таблиц справочников и 10 – для таблиц заказов.

Дополнительные таблицы

«Города»

Наименование поля	Тип (формат) поля
Код_города	Счетчик
Наименование	Текстовой

«Сотрудники»

Наименование поля	Тип (формат) поля
Код_сотрудника	Счетчик
ФИО	Текстовой
Телефон	Текстовой

Дополнительные атрибуты

Но-мер	Наименование поля	Тип (формат) поля
1	2	3
Для таблицы «Фирма»		
1	Код_города	Числовой (связь с таблицей «Города»)
2	ФИО_директора	Текстовой
3	Телефон	Текстовой
4	E-mail	Текстовой
5	Расчетный_счет	Текстовой
Для таблицы «Товар»		
6	Дата_поставки	Дата/время
7	Годен_до	Дата/время

1	2	3
8	Измерение	Текстовой
9	Производитель	Текстовой
Для таблицы «Заказ_фирмы»		
10	Дата_выполнения	Дата/время
11	Код_отв_сотрудника	Числовой (связь с таблицей «Сотрудники»)

Таблица вариантов

№ вар.	1 (+создание таб. «Города»)	2	3	4	5	6	7	8	9	10	11 (+создание таб. «Сотр.»)
1	*										
2		*						*			
3			*						*		
4				*						*	
5					*	*					
6				*			*				
7			*					*			
8		*							*		
9					*					*	
10											*
11			*				*				
12				*				*			
13					*		*				
14		*								*	
15					*			*			

Контрольные вопросы

1. Какие варианты создания таблиц в Access вы знаете?
2. Что такое схема данных?
3. Какие типы данных используются в Access?
4. Как устанавливаются связи между таблицами в Access?
5. Как обеспечивается целостность данных в Access?

ЛАБОРАТОРНАЯ РАБОТА № 2

Создание форм и страниц доступа к данным

Цель работы. Необходимо разработать ряд пользовательских форм, для работы с приложением для созданной базы данных. Для размещения приложения по работе с БД в сети использовать страницы доступа к данным.

Форма – объект приложения БД, предназначенный для ввода и редактирования данных. Для работы с формами приложения служит закладка *Формы* списка *Объекты*.

Контрольный пример

1. Создание форм с помощью *Мастера форм*.

Для создания формы для таблицы **Фирмы** воспользуемся *Мастером* форм. Переходим на закладку **Формы**, выбираем кнопку *Создать*. В появившемся окне (рис. 6) указываем способ создания (*Мастер форм*) и таблицу, для которой создается форма (**Фирма**). В следующем окне выбираем поля таблицы, которые будут отображаться на форме. Поскольку поле *Код фирмы* имеет тип *Счетчик*, то значение его генерируется автоматически и на форме поле может не присутствовать.

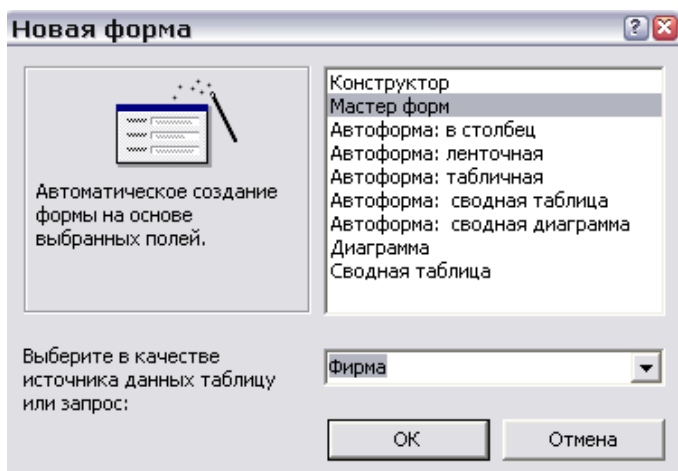


Рис. 6. Создание форм

В следующем окне укажем внешний вид формы по желанию, далее – зададим стиль оформления и наконец назовем форму и нажмем кнопку *«Готово»*. Результат представлен на рис. 7.

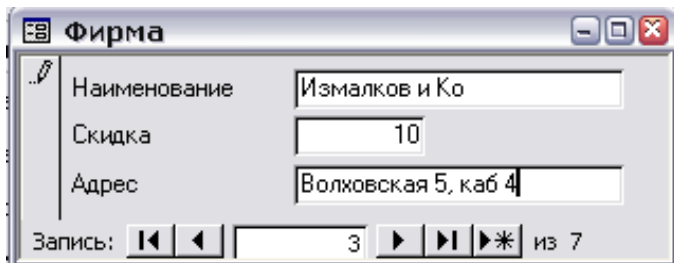


Рис. 7. Форма **Фирма**

2. Создание форм в режиме *Конструктора*

Формы Товары создаем с помощью Конструктора: при указании способа создания выбираем пункт *Конструктор*, в качестве таблицы выберем **Товар**. С помощью контекстного меню добавляем области *заголовка* и *примечания* формы.

Используя *Панель элементов для конструирования* (рис. 8), разместим на новой форме необходимые компоненты.

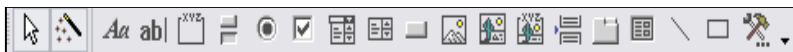


Рис. 8. Панель элементов

1. Добавим заголовок с помощью элемента *Надпись*.

2. Из схемы данных таблицы **Товар** (дочернее окно на окне конструктора) перетащим на форму необходимые поля. Код можно не добавлять, т.к. коды товара генерируются автоматически (тип поля – счетчик).

При необходимости изменим надпись перед *Поле для ввода данных*, добавим примечание, откорректируем размеры элементов.

Если первоначально при создании не была указана таблица-источник, это можно сделать, откорректировав свойства формы (пункт меню Вид ~ Свойства). В поле *Источник записей* закладки *Данные* нужно указать требуемую таблицу. В этом же окне можно откорректировать и другие свойства формы, например, изменить её цвет или убрать навигатор работы с записями (Макет ~ Кнопки перехода ~ Нет). В итоге получим форму, аналогичную рис. 9.

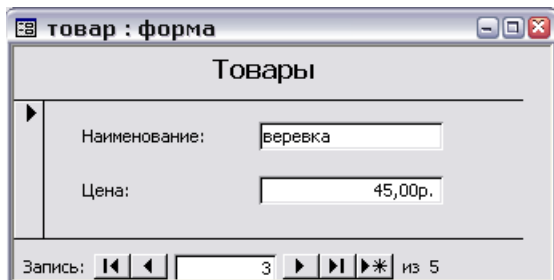


Рис. 9. Форма **Товар**

3. Разработка многотабличных форм

Составная многотабличная форма создается для работы с несколькими взаимосвязанными таблицами. Многотабличная форма может состоять из основной части и одной или нескольких подчиненных включаемых форм, т.е. быть составной. Подчиненная форма может быть построена на основе, как подчиненной, так и главной таблицы относительно таблицы-источника основной части формы.

1. Для создания формы в окне базы данных выберем закладку *Формы* и нажмем кнопку *Создать*.

2. В окне *Новая форма* выберем режим создания *Мастер форм* и выберем в качестве источника данных основной части формы из списка таблицу **Заказ-фирма**.

3. В открывшемся диалоговом окне *Создание форм* сначала выбираем таблицу **Заказ_фирма** и поля из неё, включаемые в форму. После этого в этом же окне выберем таблицу **Заказ-товар** и укажем необходимые поля (код товара и количество) рис. 10.

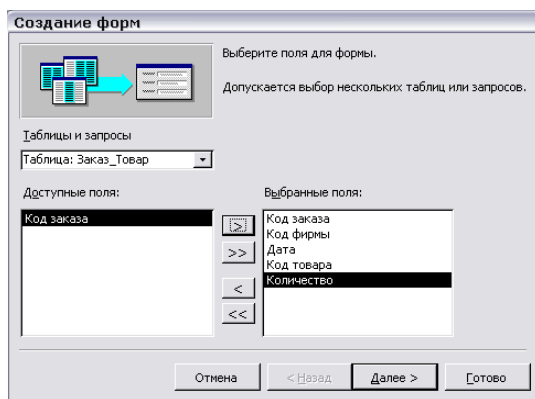


Рис. 10. Окно для выбора полей формы

4. Перейдите на следующее окно, нажав кнопку Далее.

5. В окне создания форм выделим таблицу **Заказ_фирма**, которая будет являться источником основной части формы и выбираем один из вариантов подключения формы (в нашем случае лучше выбрать вариант Подчиненные формы).

6. В следующем окне выберем вариант оформления подчиненной формы.

7. Выберем стиль оформления формы на следующем окне.

8. В последнем диалоговом окне Создание форм отредактируем заголовки формы.

9. Доработаем форму в режиме конструктора:

9.1. дадим форме заголовок;

9.2. уберем автоматический счетчик записей на основной форме (см. Создание форм в режиме конструктора) и поместим на форму кнопки для перемещения по записям, редактирования записей.

Для этого перетаскиваем элемент *Кнопка* с панели инструментов, в появившемся окне (рис. 11) определяем Категорию и Действие, которое будет выполняться при нажатии на кнопку (например действие *Добавить запись* категории *Обработка записей*), выбираем рисунок кнопки (или подпись) и даем ей имя. Аналогично можно создать кнопки для обработки других событий, например перехода по записям.

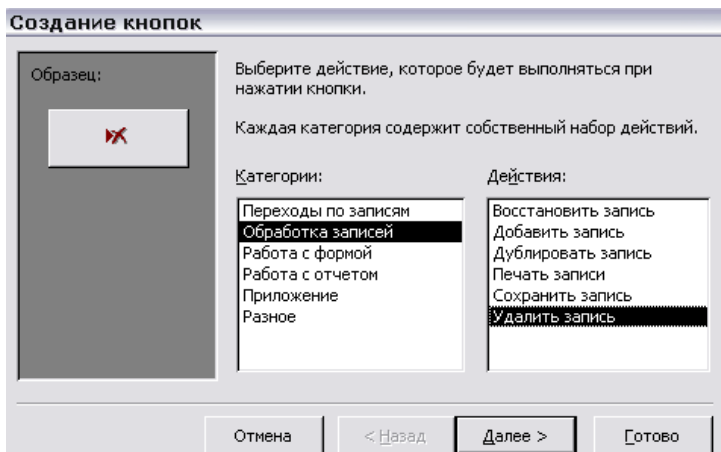


Рис. 11. Задание процедур кнопок

Результатом работы должна стать форма следующего вида рис. 12.

Рис. 12. Многотабличная форма

4. Создание страницы доступа.

Страница доступа к данным представляет собой web-страницу, которую можно использовать для добавления, редактирования, просмотра текущих данных в БД Access.

Создадим страницу доступа для работы с информацией о фирмах, на основе соответствующей формы.

Для этого выберем форму *Фирмы* и пункт меню *Файл ~ Сохранить как*. В открывшемся окне дадим имя новому файлу и укажем Как: «Страница доступа к данным». Файл желательно сохранить в той же папке, что и файл базы данных. Итогом выполнения этих операций станет файл с расширением *.html. На полученной странице размещаются элементы формы (поля, заголовки, навигатор).

Созданная страница будет отображаться на закладке *Страницы* окна базы данных. Такую же страницу можно было получить, перейдя на эту вкладку и нажав *Создать*. Страницы можно создавать в режиме *Мастера* или *Конструктора*.

Откорректируем созданную страницу с помощью *Конструктора*: изменим заголовок и оформление, добавим пользовательские кнопки (создаются так же, как на формах). Для настройки связи страницы с базой данных используются *Свойства подключения для страницы* (вкладка *Подключение*) окна *Список полей*. В соответствующем поле прописывается/изменяется путь к БД.

Результат создания страницы представлен на рис. 13.

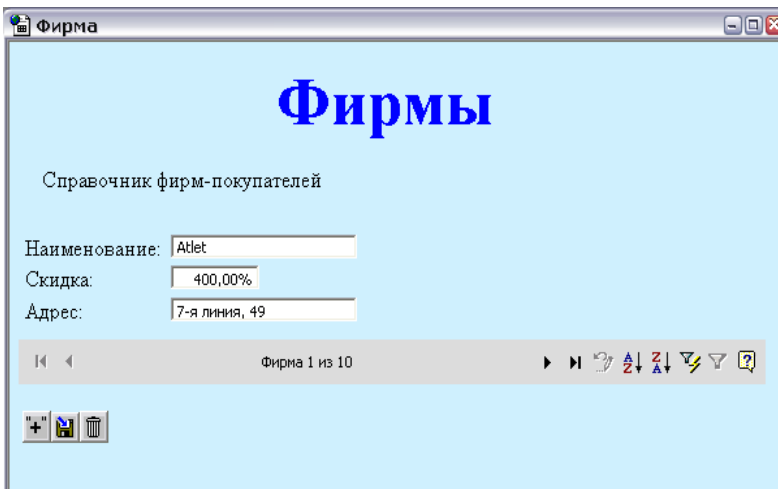


Рис. 13. Страница доступа по работе с фирмами

Индивидуальное задание

1. Для вариантов 1 и 10 создать дополнительные формы по работе с таблицами городов или сотрудников. Все остальные варианты включают свои дополнительные поля на существующие формы.

2. Для четных вариантов добавить на формы со стандартными навигаторами кнопки для **удаления** записей (стандартный интерфейс их не содержит). Для нечетных вариантов – добавить кнопки **поиска** произвольной записи из таблицы.

3. Добавить в таблицы ещё по 5 записей с помощью созданных форм.

Контрольные вопросы

1. С какой целью создаются формы в Access?
2. Какие способы создания форм существуют в Access?
3. Опишите кнопки панели элементов?
4. Как создаются страницы доступа к данным?
5. Опишите технологию создания и обработки кнопки на форме.

ЛАБОРАТОРНАЯ РАБОТА № 3

Обработка данных средствами Access

Цель работы. Разработать запросы к БД, используя редактор ABE (запросы по абзацу).

Одним из основных инструментов обработки данных в СУБД являются запросы. Запрос – это формализованное требование пользователя на отбор данных или на выполнение действий. Запрос строится на основе одной или нескольких таблиц. При этом могут использоваться таблицы базы данных, а также сохраненные таблицы, полученные в результате других запросов. Запрос позволяет выбрать необходимые данные из одной или нескольких взаимосвязанных таблиц, произвести вычисления и получить результат в виде таблицы. В Access может быть создано несколько видов запроса:

Запрос на выборку – выбирает данные из взаимосвязанных таблиц и других запросов. Результатом его является таблица, которая существует до закрытия запроса.

Запрос на создание таблицы – основан на запросе выборки, но в отличие от него результат запроса сохраняется в новой таблице.

Запросы на обновление, удаление, добавление – являются запросами действия, в результате выполнения которых изменяются данные в таблицах.

Основы конструирования запросов

Основные принципы конструирования запроса заложены в технике конструирования запроса на выборку, являющегося основой всех видов запроса. Разработка запроса производится в режиме *Конструктор запросов*. Для создания запроса надо в окне базы данных выбрать закладку *Запросы* (пункт *Запросы* меню *Объекты*) и нажать кнопку *Создать*. В открывшемся окне *Новый запрос* выбрать *Конструктор*. В окне *Добавление таблицы* выбрать используемые в запросе таблицы и нажать кнопку *Добавить*. Затем кнопкой *Закрыть* выйти из окна *Добавление таблицы*. В результате появится окно конструктора запросов. Окно конструктора разделено на две панели. Верхняя панель содержит схему данных запроса, а нижняя панель является бланком запроса по образцу.

Схема данных запроса отображает выбранные таблицы и связи между ними, имеющиеся в схеме данных БД. Бланк запроса по образцу представлен в виде таблицы в нижней панели окна запроса. Каждый столбец бланка относится к одному полю, с которым нужно работать в запросе.

При заполнении бланка необходимо:

- В строку *Поле* включить имена полей, используемые в запросе
- В строке *Вывод на экран* отметить поля, которые должны быть включены в результирующую таблицу
- В строке *Условие отбора* задать условия отбора записей
- В строке *Сортировка* выбрать порядок сортировки записей результата

Если в запрос включаются все поля таблицы, то в строке Поля ставится знак *.

Контрольный пример

Технология конструирования запросов

1. Рассмотрим технологию создания *однотабличного запроса на выборку* на примере получения информации из таблицы **Товар** базы данных **Учет заказов**.

• Пусть необходимо получить информацию о *товаре* с названием *Коврик*.

1) Для создания запроса в окне базы данных выберем закладку *Запросы* и нажмем кнопку *Создать*.

2) В окне *Новый запрос* выберем *Конструктор*.

3) В окне *Добавление таблицы* выберем таблицу **Товар**, нажмем кнопку и закроем окно *Добавление таблицы*.

4) Заполним бланк запроса по образцу в соответствии с рис. 14.

Поле:	Наименование	Цена	
Имя таблицы:	Товар	Товар	
Сортировка:			
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Условие отбора:	"Коврик"		
или:			

Рис. 14. Бланк запроса по образцу

5) Закроем *Конструктор* и сохраним запрос с именем **Коврики**. Выполним сохраненный запрос нажатием кнопки *Открыть*. В результате имеем таблицу (рис. 15).

Наименование	Цена
коврик	3 000,00р.
	0,00р.

Запись: 2 из 2

Рис. 15. Результат выполнения запроса **Коврики**

• Пусть необходимо получить информацию о всех товарах из таблицы **Товар** с ценой больше 30р.

Для этого повторим пункты 1–3 предыдущего запроса (создание), затем заполним бланк запроса по образцу в соответствии с рис. 16.

Закроем *Конструктор* и сохраним запрос с именем **Цена товара**.

Товар

*
Код товара
Наименование
Цена

Поле: Наименование Цена
Имя таблицы: Товар Товар
Сортировка:
Вывод на экран:
Условие отбора: >30
или:

Рис. 16. Запрос на выборку товара с ценой >30

Выполним сохраненный запрос. В результате имеем таблицу (рис. 17.).

Наименование	Цена
табуретка	152,02р.
веревка	45,00р.
коврик	3 000,00р.
	0,00р.

Запись: 4

Рис. 17. Результат выполнения запроса **Цена товара**

- Пусть необходимо выбрать фирмы, названия которых начинаются с буквы «А» из таблицы **Фирмы**.

1) Выполним пункты 1–3 предыдущих запросов (выбрать таблицу **Фирма**).

2) Заполним бланк запроса по образцу в соответствии с рис. 18.

Фирма

*
Код фирмы
Наименование
Скидка
Адрес

Поле: Наименование Адрес Скидка
Имя таблицы: Фирма Фирма Фирма
Сортировка:
Вывод на экран:
Условие отбора: Like "А*"
или:

Рис. 18. Запрос на выборку фирм

3) Закроем *Конструктор* и сохраним запрос с именем **Фирмы на А**. Выполним сохраненный запрос нажатием кнопки *Открыть*. В результате имеем таблицу (рис. 19.).

Наименование	Адрес	Скидка
Алертные слоники	11-я линия, 48	5
Адвентизм и Ко	Ленина, 46	8
		0

Запись: 3 из 3

Рис. 19. Результат выполнения запроса на выборку фирм

2. Рассмотрим технологию конструирования *многотабличного запроса на выборку* на примере получения информации из таблиц **Заказ_Фирма**, **Заказ_Товар** и **Товар** базы данных **Учет заказов**.

- Пусть необходимо получить информацию о товарах, проданных в период с 1.01.2002 до 1.01.2003.

1) Повторим пункты 1–3 предыдущих запросов.

2) В окне *Добавление таблицы* выберем таблицы **Заказ_Фирма**, **Заказ_Товар** и **Товар** и закроем окно *Добавление таблицы*. Связи между таблицами установлены автоматически в соответствии со схемой данных базы данных.

3) Заполним бланк запроса по образцу в соответствии с рис. 20.

Поле:	Наименование	Цена	Дата
Имя таблицы:	Товар	Товар	Заказ_Фирма
Сортировка:			
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Условие отбора:			>#01.01.2002# And <#01.01.2003#
или:			

Рис. 20. Запрос на выбор товара по датам

4) Закроем *Конструктор* и сохраним запрос с именем **Товары по дате**. Выполним сохраненный запрос нажатием кнопки *Открыть*. В результате имеем таблицу (рис. 21).

Наименование	Цена	Дата
▶ мыло	4,00р.	12.12.2002
веревка	45,00р.	13.07.2002
коврик	3 000,00р.	15.03.2002
табуретка	152,02р.	15.03.2002
сикатор	10,52р.	24.10.2002
*		

Запись: 1 из 5

Рис. 21. Результат выполнения запроса Товары по дате

3. Рассмотрим технологию конструирования многотабличного запроса на выборку с использованием вычисляемых полей.

- Пусть необходимо получить информацию о заказах, фирмах, сделавших заказ и суммах по оплате каждого товара.

1) Повторим пункты 1–3 предыдущих запросов.

2) В окне *Добавление таблицы* выберем таблицы **Заказ_Фирма**, **Заказ_Товар**, **Фирма** и **Товар** и закроем окно *Добавление таблицы*.

3) В рассматриваемом запросе поле **Сумма** является вычисляемым и в таблицах не присутствует. Для его создания в бланке запроса по выбору необходимо в строке *Поле* указать *Имя нового поля* – *Сумма* и через двоеточие выражение, с помощью которого будут вычисляться значения:

Сумма: Заказ_товар!Количество×Товар!Цена×(1-Фирма!Скидка/100)
(деление на 100 необходимо, если скидка изначально не задана в процентном формате).

Для построения выражения можно воспользоваться строителем выражений (а можно ввести вручную). Для его вызова необходимо курсор установить в строке *Поле*, щелкнуть правой клавишей мыши и в контекстном меню выбрать пункт *Построить*. В появившемся окне (рис. 22.) строим выражение, выбирая таблицу в окошке слева, нужное поле в центральном списке и <значение> справа и нажимая кнопку *Вставить*. Потом добавляем скобки или знак операции с помощью кнопок или с клавиатуры, затем следующее выражение и т.д. Фиксируем выражение нажатием кнопки *ОК*.

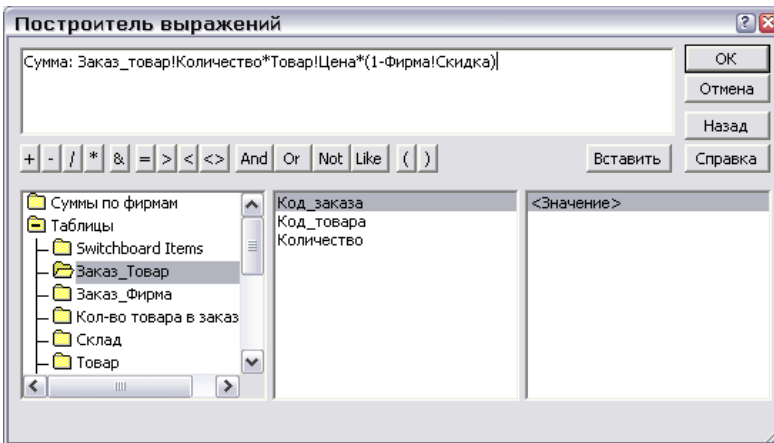


Рис. 22. Окно построителя выражений

4) Заполним бланк запроса по образцу в соответствии с рис. 23.

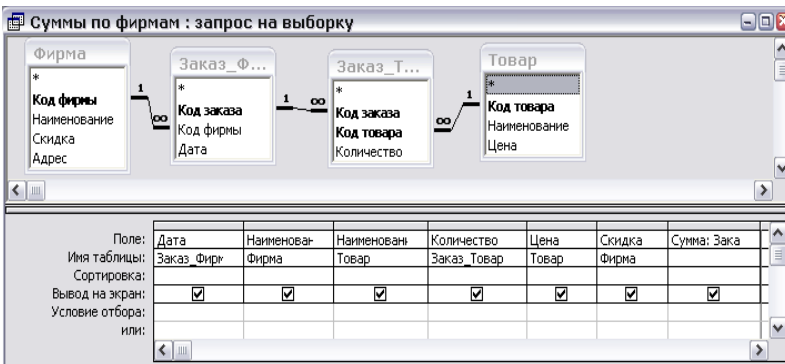


Рис. 23. Запрос с использованием вычисляемого поля

5) Закроем *Конструктор* и сохраним запрос с именем **Суммы по фирмам**. Выполним сохраненный запрос. В результате имеем таблицу (рис. 24).

	Дата	Фирма.Наименов	Товар.Наименов	Количество	Цена	Скидка	Сумма
▶	14.03.2003	Маня и Ко	лезвие бритвы	120	28,00р.	5	8192
	12.12.2002	Варя и Ко	мыло	340	4,00р.	3	1319,2
	14.03.2003	Маня и Ко	табуретка	50	152,02р.	5	7220,95
	13.01.1998	Кикимора+	сикатор	50	10,52р.	2	515,48
	12.03.2003	Кикимора+	веревка	40	45,00р.	2	1764
	13.07.2002	Маня и Ко	веревка	220	45,00р.	5	9405
	12.03.2003	Кикимора+	мыло	67	4,00р.	2	262,64
	12.03.2003	Кикимора+	лезвие бритвы	78	28,00р.	2	2140,32
	15.03.2002	Измалков Inc.	коврик	2	3 000,00р.	10	5400
	15.03.2002	Кикимора+	табуретка	10	152,02р.	2	1489,796
	14.10.2003	Маня и Ко	сикатор	3	10,52р.	5	29,88

Запись: 1 из 11

Рис. 24. Результат выполнения запроса Суммы по фирмам

Индивидуальное задание

1. Для вариантов 1 и 10 создать запрос по образцу на выборку любой фирмы из таблицы **Фирма** (по наименованию) с точным совпадением значения. Для вариантов 2,8 и 14 запрос на выборку по значению ФИО директора, для вариантов 3,7,11 – по значению телефона фирмы, для № 4,6,12 – по значению E-mail фирмы, для № 5,9,13,15 – по значению расчетного счета фирмы.

2. Для четных вариантов выполнить запрос на выборку фирм со скидкой не больше определенного значения, для нечетных – со скидкой меньшей определенного значения.

3. Для вариантов 1 и 10 создать запрос по образцу на выборку городов или сотрудников, название (имя) которых начинается на определенную букву. Для вариантов 2,3,7,8,12,15 – выполнить запрос на выборку товаров по первой букве производителя или единицы измерения в зависимости от имеющегося атрибута. Вариантам 4,5,6,9,11,13,14 – создать запрос на выборку товара, не начинающегося на определенную букву.

4. Для всех вариантов выполнить запрос на информацию о фирмах, приобретающих определенный вид товара (добавить в запрос 4 основные таблицы, в качестве условия записать конкретное наименование имеющегося (!) товара, это поле на экран можно не выводить).

5. Для всех вариантов на основе запроса Суммы по фирмам создать новый (а не модифицировать старый!) запрос, выбирая информацию по конкретной фирме. Совет: скопируйте запрос о суммах по фирмам стандартным образом в эту же БД и модифицируйте копию с учетом нового условия.

Контрольные вопросы

1. Какие типы запросов существуют в Access?
2. Как формируются условия отбора записей в запросе на выборку?
3. Опишите технологию создания вычисляемого поля.
4. Какие возможности предоставляет Построитель выражений?
5. Приведите примеры использования логических операций в условиях отбора?

ЛАБОРАТОРНАЯ РАБОТА № 4

Создание запроса с произвольной выборкой Корректировка данных средствами запроса

Цель работы: Научиться создавать запросы с условиями поиска, вводимыми пользователем, запросы с групповыми операциями, запросы на создание и обновление таблиц.

Запросы с произвольной выборкой

Пусть необходимо получать информацию о начисленных суммах по различным фирмам. Для решения этой задачи можно использовать два пути:

- сконструировать запрос отдельно для каждой фирмы. Это не очень эффективно, т.к. количество фирм меняется, что заставляет постоянно добавлять новые запросы.
- создать один запрос на выборку, задавая условия поиска в специально созданной форме (запрос с произвольной выборкой).

*Создавая запрос с произвольной выборкой, необходимо заранее создать опорный запрос. В качестве опорного запроса в нашем случае можно выбрать запрос **Суммы по фирмам** (см. Лабораторную работу № 3). Сделайте копию запроса **Суммы по фирмам** с именем **Суммы по фирмам 1**.*

Контрольный пример

1. Создадим *Форму* для задания критериев отбора записей в режиме *Конструктора* обычным способом. Используя *Панель элементов для конструирования*, разместим на новой форме необходимые компоненты:

Переместим на форму элемент *Поле*, изменим надпись перед *Поле*: удалим условное название «Поле» и введем комментарий «Введите название фирмы».

Выберем элемент *Кнопка* и переместим его на форму.

В открывшемся окне *Создание кнопок* выберем категорию *Разное* и *Действие* **выполнить запрос**.

Нажмем кнопку *Далее* и выберем запрос, который откроется при нажатии на эту кнопку – **Суммы по фирмам**.

Далее выберем оформление для кнопки (Текст или рисунок).

В результате выполненных действий форма для запроса примет вид (рис. 25.).

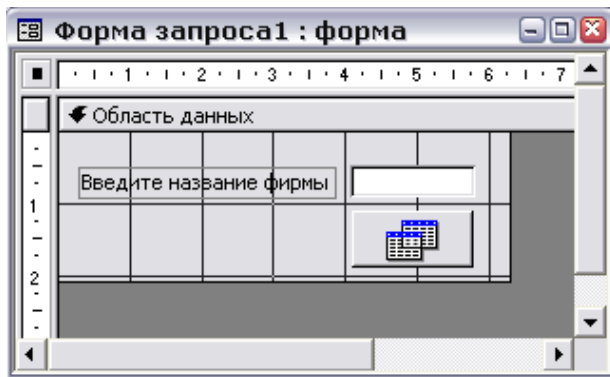


Рис. 25. Форма для запроса

Остается закрыть *Конструктор* и дать новой форме имя **Форма запроса1**.

2. В связи с тем, что условия для выборки будут задавать из формы, необходимо преобразовать Запрос **Суммы по фирмам**.

Для этого откроем запрос **Суммы по фирмам** в режиме *Конструктора*.

В поле **Наименование. Фирма** (строку *Условие отбора*) ввести выражение:

Like [Forms]![Форма запроса1]![Поле0]

Выражение рекомендуется вводить с помощью построителя выражений (см. Лабораторную работу № 3), выбирая операцию, потом тип операнда – формы, нужную форму в списке, и требуемый элемент (Поле) из имеющихся и нажимая кнопку *Вставить*. Закроем запрос, сохранив изменения.

Откроем закладку *Формы* и откроем **Форму запроса1**. Введем условие для отбора записей (рис. 26.).

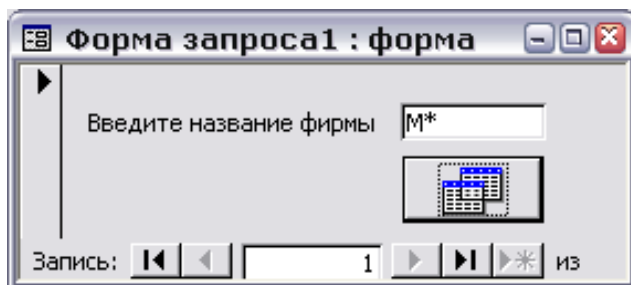


Рис. 26. Форма для запроса с условием для отбора записей

Результат выполнения запроса представлен на рис. 27.

Дата	Фирма.Наименован	Товар.Наименован	Количество	Цена	Скидка	Сумма
14.03.2003	Маня и Ко	лезвие бритвы	120	28,00р.	5	3192
14.03.2003	Маня и Ко	табуретка	50	152,02р.	5	7220,95
13.07.2002	Маня и Ко	веревка	220	45,00р.	5	9405
24.10.2002	Маня и Ко	сикатор	3	10,52р.	5	29,982

Рис. 27. Результат выполнения запроса на выборку

Корректировка данных средствами запроса

1. Использование групповых операций в запросах

Групповые операции позволяют выделить группы записей с одинаковыми значениями в указанных полях и использовать для других полей этих групп определенную статистическую функцию.

Пусть требуется создать запрос для таблицы **Заказ-Товар**, где количество для одинаково наименования товара суммируется.

1) Создадим запрос на выборку для таблицы **Заказ-Товар**. Из списка таблицы **Заказ-Товар** перенесем в бланк запроса поля **Код товара**, по которому должна производиться группировка и **Количество товара**, будем использовать функцию **Sum** для подсчета количества товара.

2) Нажмем на кнопку *Групповые операции* (при отсутствии строки *Групповые операции* выполнить команду Вид – *Групповые операции*. Заменим слово *Группировка* в столбце **Кол-во товара** на функцию **Sum** (рис. 28.).

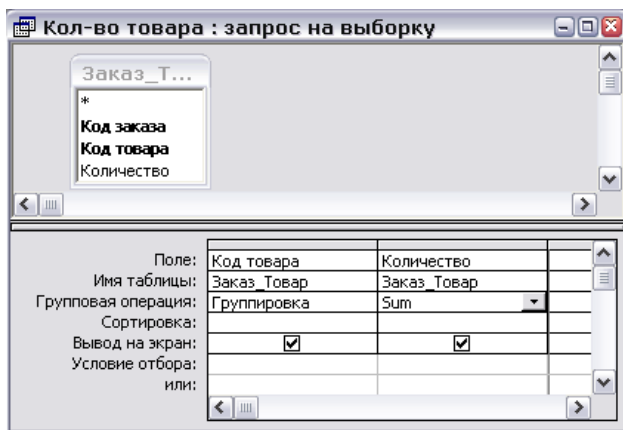


Рис. 28. Запрос с групповой операцией

Сохранить запрос с именем **Кол-во товара**.

3) Откроем запрос на выполнение. Убедитесь, что в таблице просуммированы данные по каждому из товаров в заказах.

2. Конструирование запроса на создание таблицы

Запрос на создание таблицы используется для сохранения результата запроса. Этот вид запроса основан на *запросе на выборку*, но в отличие от него сохраняет таблицу с результатами запроса. Необходимость в сохранении результатов запроса возникает, например, когда невозможно построить запрос непосредственно на другом запросе.

Сформируем запрос на создание таблицы, используя запрос **Кол-во товара**. Для этого выполним следующие действия в режиме *Конструктор*:

1) преобразуем этот запрос в запрос на создание таблицы. Выбирается пункт меню **Запрос ~ Создание таблицы**.

2) В окне **Создания таблицы** введем имя создаваемой таблицы – **Кол-во товара в заказе**. Закроем запрос. Обратите внимание на изменение пиктограммы запроса **Количество товара**.

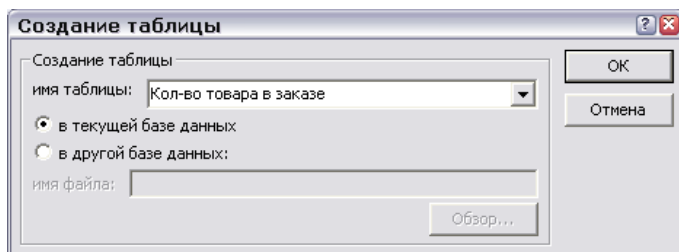


Рис. 29. Определение имени таблицы, создаваемой в запросе

3) Запустите модифицированный запрос **Количество товара**.

4) Передайте на вкладку **Таблица** и убедитесь в создании новой таблицы «**Количество товара в заказе**»

3. *Конструирование запроса на обновление*

При выполнении этого запроса изменения вносятся в группу записей, отбираемых с помощью указанных пользователем условий отбора. Значения для изменений в полях определяются в бланке запроса в строке *Обновление*.

1) Добавим в БД **Учет заказов** – таблицу **Склад**.

Наименование полей	Тип	Примечание
Код склада	Числовой	Ключевое поле
Код товара	Числовой	
Получено	Числовой	
Остаток	Числовой	

Настроить связи, сделать подстановку нужных полей. Заполнить таблицу записями для всех товаров. При этом значение поля **Получено** должно быть больше значения поля **Остаток** и оба должны быть положительными.

2) Создадим запрос на обновление на примере обновления поля **Остаток** в таблице **Склад**. Остаток высчитывается по формуле:

Получено на складе – Кол-во товара в заказе

Количество товара в заказе получено в запросе на выборку **Кол-во товара** с использованием функции *Sum*.

!Запрос на обновление непосредственно на основе запроса построить нельзя!

Поэтому используется для обновления не запрос, а таблица **Кол-во товара в заказе**, полученная по запросу на создание таблицы.

Для формирования запроса на обновление сначала создается запрос на выборку на основе трех таблиц – обновляемой таблицы **Склад**, таблицы **Товар** и таблицы **Кол-во товара в заказе**. Для преобразования запроса на выборку в запрос на обновление выбирается пункт меню **Запрос ~ Обновление**.

Перетащить поле **Остаток** из списка таблицы **Склад**. В строке *Обновление* ввести формулу:

[Получено]-[Sum-Количество]

Сохранить запрос под именем **Обновление Склада**.

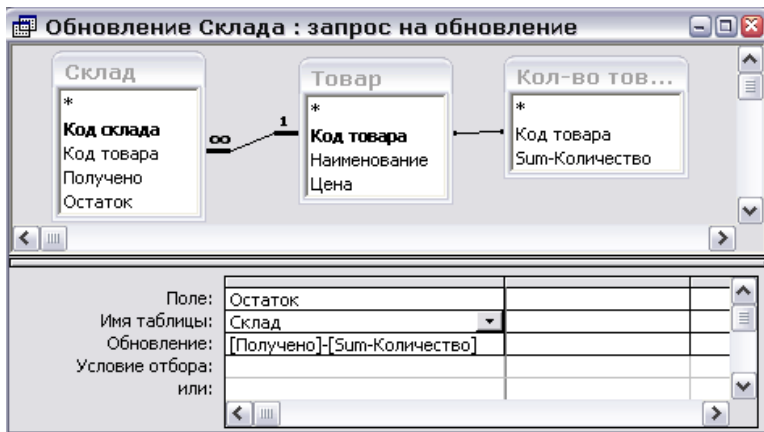


Рис. 30. Запрос на обновление таблицы Склад

Выполненные нами запросы на создание и обновление должны работать по следующей схеме: сначала вносятся данные в таблицы **Заказ Фирма** и **Заказ Товар**, т.е. добавляются товары в заказ. Следовательно, их необходимо со склада списать. Откроем таблицу **Склад**, посмотрим на столбец **Остаток**, запомним значения. Для списания товара вызываем запрос на создание таблицы **Количество товара**, если запрос уже выполнялся и таблица уже существует, подтвердим её замену. После этого запускаем запрос **Обновление склада**, при этом СУБД, учитывая данные только что измененные данные в таблице **Количество товара**, изменит значения остатков в таблице **Склад**. Убедиться в этом можно, открыв таблицу **Склад** и сравнив новые значения с запомненными старыми.

Индивидуальное задание

1. Для четных вариантов выполнить запрос с произвольной выборкой на отбор товара по первой букве наименования (создать форму и запрос).
2. Для нечетных вариантов – запрос с произвольной выборкой из таблицы **Склад** товаров, значение остатка которых меньше вводимого пользователем (создать форму для ввода барьера остатка и запрос).

Контрольные вопросы

1. Как выполняется корректировка данных средствами запроса?
2. Для чего используются групповые операции в запросах?
3. Опишите технологию создания запроса для создания таблицы.
4. Как происходит обращение к элементам формы в запросах?
5. Как сформировать запрос с параметрами?

ЛАБОРАТОРНАЯ РАБОТА № 5

Редактирование базы данных средствами VBA

Цель работы. Разработать форму для работы с заказами, учитывая выбор клиента и заказываемых им товаров, внесение в базу текущей даты заказа, наличие товаров на складе и списание их при оформлении заказов. Реализованы эти функции должны быть в виде процедур обработки событий объектов формы, написанных на языке VBA (Visual Basic for Application).

Контрольный пример

1. Создаем макет формы заказов.

Создаем новую форму в режиме *Конструктора*. (Формы ~ Создать ~ Конструктор), с помощью контекстного меню добавляем заголовок и примечание формы. После этого перетаскиваем на форму *поле со списком*, при этом должны быть включены мастера (иконка «Волшебная палочка» панели инструментов). В окне создания списка (рис. 31.) указываем, что данные будут браться из таблицы, в следующем окне мастера выбираем эту таблицу.

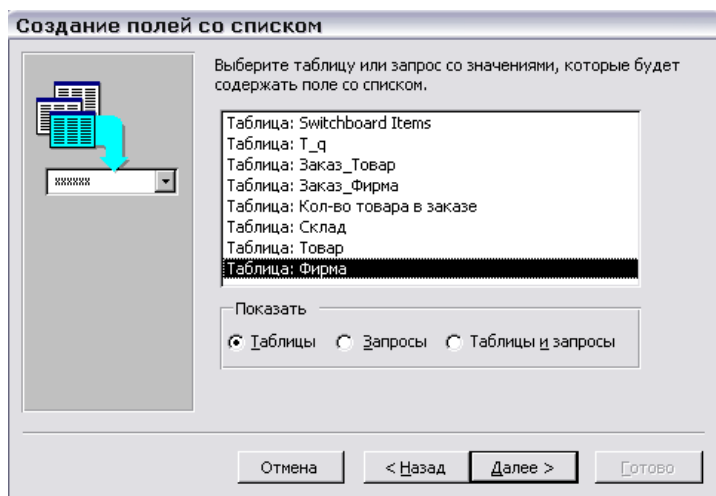


Рис. 31. Мастер создания полей со списком

Поскольку необходимо выбирать фирму-заказчика, то указываем таблицу **Фирма**. Затем выбираем нужное поле – **Наименование**, настраиваем ширину списка, даем ему имя и нажимаем на *Готово*. После этого на форме корректируем сопроводительную надпись списка. Ана-

логичным образом формируем список для наименований товаров (таблица **Товар**).

Кроме этого, необходимо добавить на форму пять полей с подписями и четыре кнопки, причем две из них можно сделать с помощью мастера (кнопки *Отмена* и *Добавить фирму*). Для первой кнопки после перетаскивания кнопки на форму в окне *Создания кнопок* указываем категорию *Работа с формой* и действие – *Закрывать форму*. Для добавления фирмы необходимо просто открыть форму **Фирма**, созданную ранее, где и добавить новую запись (категория – *Работа с формой*, действие – *Открыть форму*). Для остальных кнопок будут написаны процедуры обработки. Измените свойства полей предназначенных для вывода остатка и цены на товары в соответствии с рис. 32.

Добавим на форму группировочные рамки и линии оформления, в итоге получим форму рис. 32.

Рис. 32. Макет формы работы с заказами

2. Схема работы с формой

Подразумевается, что работа пользователя будет проходить по следующему алгоритму:

1) Выбирается фирма из списка. При её отсутствии пользователь переходит на форму фирмы с помощью кнопки *Добавить фирму* и добавляет клиента.

2) Нажатием кнопки *Новый заказ* пользователь формирует новую запись в таблице заказов. Программа отображает номер заказа.

3) Выбирается товар из списка. Система отображает его цену и количество на складе. Если эти параметры удовлетворяют пользователя, он вносит необходимое для заказа количество в соответствующее поле и нажимает кнопку *Добавить в заказ*. При этом заказанное количество товара со склада списывается.

4) Повторяется шаг 3 до тех пор, пока все товары не будут добавлены в заказ. После добавления каждого из них система пересчитывает общую стоимость заказа в соответствующем поле.

3. Отображение информации на форме в соответствии с выбранным значением списка.

Очевидно, что при выборе товара из списка в поле *Остаток* должно высвечиваться имеющееся количество его на складе, а в поле *Цена* – цена из таблицы **Товар**. Для этого напишем процедуры обработки такого события, как *Изменение* поля со списком. Процедуры пишутся в редакторе VBA, вызов его осуществляется следующим образом. 1) Вызываем с помощью контекстного меню окно свойств для поля со списком товаров (щелкаем по полю правой кнопкой мыши, выбираем пункт *Свойства*). 2) Выбираем вкладку *События*, для пункта *Изменение* выбираем [*Процедура обработки*] и нажимаем на [...] рядом. В появившемся окне (основной части) представлены все процедуры нашей формы. Конкретно же процедура, связанная с изменением списка, должна выглядеть примерно следующим образом: ***Не забудьте учесть, что номера кнопок, полей и полей со списком, названия переменных у вас могут быть другими!!!! Это касается всех процедур приведенных ниже!!!!***

```
Private Sub ПолеСоСписком29_Change()  
    Dim b As Integer // заводим целочисленную переменную  
    Dim rs As Recordset // заводим переменную типа запись  
    b = ПолеСоСписком29.Value // присваиваем переменной значение  
    из списка товаров  
    Set rs = CurrentDb.OpenRecordset("SELECT * FROM Товар WHERE  
Код_товара=" + Str(b))  
    // формируем динамический sql-запрос на выбор из таблицы това-  
ров товара с нужным кодом  
    Поле16 = rs!Цена // считываем в поле значение цены, выбранного  
товара  
    Set rs = CurrentDb.OpenRecordset("SELECT * FROM Склад WHERE  
Код_товара=" + Str(b))  
    // аналогично для количества со склада  
    Поле14 = rs!Остаток  
End Sub
```

Необходимо также учесть, что при открытии формы поля **Цена** и **Остаток** должны быть очищены. Для этого в окне свойств *формы* выбираем *События ~ Открытие ~ [Процедура обработки]* и формируем следующую или похожую процедуру:

```
Private Sub Form_Open(Cancel As Integer)
Поле14.SetFocus
Поле14.Text = ""
Поле16.SetFocus
Поле16.Text = ""
End Sub
```

4. Добавление записей в таблицы программным способом.

Кнопка *Новый заказ* связана с процедурой добавления в таблицу **Заказ_Фирма** новой записи. При этом код заказа генерируется автоматически (поле типа Счетчик), код клиента определяется по значению поля со списком фирм, а дата ставится текущая. Обработаться будет нажатие кнопки. Для организации этой процедуры в окне свойств кнопки *Новый заказ* выбираем *События ~ Нажатие кнопки ~ [Процедура обработки]* ~ [...]. В окне редактора вводим следующую процедуру:

```
Private Sub Кнопкаб_Click()
Set rs = CurrentDb.OpenRecordset("SELECT * FROM Заказ_Фирма")
//с помощью динамического sql-запроса открываем нужную таблицу
rs.AddNew // добавляем новую запись
rs!Код_фирмы = ПолеСоСписком27 //в поле Код_Фирмы вносим
значение из поля со списком фирм
rs!Дата = Date //в поле даты вводим текущую дату
(функция Date())
rs.Update //обновляем запись
rs.Bookmark = rs.LastModified
Поле31 = rs!Код_заказа //отображаем код нового заказа сге-
нерированный автоматический в по-
ле формы.

End Sub
```

Кнопка *Добавить в заказ* добавляет записи в таблицу **Заказ_Товар**, отвечает за списание заказанного количества со склада и подсчет общей стоимости заказа. Код заказа добавляется из поля, в которое предыдущая процедура внесла код текущего заказа.

```
Private Sub Кнопка18_Click()
Dim b As Integer
Dim rs As Recordset
b = ПолеСоСписком29.Value //присваиваем переменной значение
поля со списком (код товара)
```

```

Set rs = CurrentDb.OpenRecordset("SELECT * FROM Склад WHERE
Код_товара=" + Str(b))
// находим в таблице склад строку нужного товара с помощью SQL-
запроса.
rs.Edit //включаем режим редактирования записи
rs!Остаток = rs!Остаток – Поле10 //изменяем остаток товара на ве-
личину заказанного количества
rs.Update //обновляем запись
Set rs2 = CurrentDb.OpenRecordset("SELECT * FROM Заказ_Товар")
//открываем таблицу Заказ_Товар
rs2.AddNew //добавляем запись
rs2!Код_заказа = Поле31 //вносим в поле код заказа значение номе-
ра заказа, сгенерированного в результате предыдущей процедуры
rs2!Код_товара = ПолеСоСписком29 //вносим код товара из поля со
списком товаров
rs2!Количество = Поле10 //вносим количество из поля формы
rs2.Update //обновляем запись
rs2.Bookmark = rs.LastModified
Sum = Sum + Val(Поле10.Value) * Val(Поле16.Value) //считаем
стоимость как произведение цены на количество товара, накапливая её
для всех товаров (***)
Поле33 = Sum //выводим значение стоимости
Поле10.SetFocus
Поле10.Text = "" //очищаем поле количества
End Sub

```

(***) *Примечание:* чтобы переменная Sum не обнулялась после каж-
дого выполнения процедуры и стоимость накапливалась, необходимо
объявить эту переменную как глобальную. Для этого в разделе (General)
добавляем строку:

```
Public Sum As Integer
```

5. Сохраните полученную форму.

6. Запустите форму и сформируйте новый заказ.

Индивидуальное задание

1. Для четных вариантов изменить код так, чтобы при попытке за-
казать количество товара большее имеющегося на складе система выда-
вала предупреждение, и данные не вносились.

2. Для нечетных вариантов добавить поле **Итого стоимость со
скидкой** и рассчитать эту стоимость, т.е. для описанной функции Sum
учесть скидку фирмы, сделавшей этот заказ. (Получить значение скидки
из таблицы **Фирма** и умножить Sum на (1-Скидка/100), т.к. скидка вы-
ражена в %).

Контрольные вопросы

1. Как описываются переменные в VBA?
2. Для чего используется тип переменной Recordset?
3. Какая команда служит для добавления записей в базу данных?
4. Как средствами VBA вывести на форме значения полей таблицы?
5. Как очистить поля формы командами VBA?

ЛАБОРАТОРНАЯ РАБОТА № 6

Создание запросов средствами языка SQL в MS Access

Цель работы: Разработать запросы к БД, используя язык SQL (структурных запросов).

Для построения запросов может быть использован язык структурированных запросов SQL (Structured Query Language), являющийся внутренним стандартом на выполнение запросов. А запросные таблицы на языке QBE (см. Лабораторную работу №3) перед выполнением преобразуются системой в выражения SQL. То есть один и тот же запрос можно создавать как в режиме Конструктора, так и непосредственно вводом инструкций SQL.

Для создания запросов средствами языка SQL необходимо:

1. Перейти на закладку *Запросы*;
2. Выбрать кнопку *Создать* и режим *Конструктор*;
3. Добавить в запрос необходимые таблицы;
4. Воспользоваться пунктом меню *Вид~Режим SQL*;
5. Набрать текст запроса.
6. Сохранить и закрыть запрос.

Структура SQL запроса на выборку:

```
SELECT [DISTINCT ROW] {*| Список полей} – список полей запроса
FROM <Список таблиц> – список участвующих таблиц
[WHERE <Условие отбора> ] – условие отбора полей
[ORDER BY <Список полей для сортировки> ] – сортировка (по
умолчанию – по возрастанию, для смены направления сортировки после
имени поля – описатель DESC)
[GROUP BY<Список полей для группирования> ]; – группировка
по определенным полям.
```

Контрольный пример

Используя приведенную технологию, создадим следующие запросы:

Запрос 1

Выбрать фирмы со скидкой больше 5%.

```
SELECT Наименование, Скидка
FROM Фирма
WHERE Скидка>5;
```

Запрос 2

Выбрать товары с наименованием на букву “м”, представив заголовки более наглядным образом. (Возможность изменять имена заголовков столбцов в запросе)

```
SELECT Наименование AS Товар, Цена AS Розничная цена
FROM Товар
WHERE Наименование Like "м*";
```

Запрос 3

Для каждого заказа вывести название сделавшей его фирмы и отсортировать заказы по дате (по убыванию).

```
SELECT Заказ_Фирма.Код_заказа, Заказ_Фирма.Дата, Фирма.Наименование
FROM Фирма, Заказ_Фирма
WHERE Заказ_Фирма.Код_фирмы=Фирма.Код_фирмы
ORDER BY Заказ_Фирма.Дата DESC;
```

Запрос 4

Получить названия фирм, купивших товар с кодом 4. Если коды товаров – текстовые поля, не забудьте взять выражение кода в кавычки.

1 вариант

```
SELECT Фирма.Наименование
FROM Фирма, Заказ_Фирма, Заказ_Товар
WHERE Заказ_Фирма.Код_заказа = Заказ_Товар.Код_заказа, Фирма.Код_фирмы = Заказ_Фирма.Код_фирмы, AND Заказ_Товар.Код_товара=4;
```

2 вариант

Строка «FROM...» формируется автоматически при добавлении таблиц в запрос. Форма «*таблица1* INNER JOIN «*таблица2* ON поле» означает связывание данных таблиц по этому полю.

```
SELECT Фирма.Наименование
FROM Фирма INNER JOIN (Заказ_Фирма INNER JOIN Заказ_Товар ON Заказ_Фирма.Код_заказа = Заказ_Товар.Код_заказа) ON Фирма.Код_фирмы = Заказ_Фирма.Код_фирмы
WHERE Заказ_Товар.Код_товара=4;
```

Запрос 5

Определить количество покупок определенного товара

```
SELECT Count(*)
FROM Товар INNER JOIN Заказ_Товар ON Товар.Код_товара=Заказ_Товар.Код_товара
WHERE Товар.Наименование= «веревка»;
```

Запрос 6

Определить общее количество товара в каждом заказе.

```
SELECT Код_заказа, SUM(Количество)
```

FROM Заказ_Товар
GROUP BY Код_заказа;

Индивидуальное задание

1. Для вариантов 2,7,12,15 выбрать товары, где значения атрибута Измерение не начинаются на определенную букву. Для вариантов 3,8 выбрать товары, где значения атрибута Производитель не начинаются на определенную букву. Для вариантов 4,6 выбрать фирмы, где значения атрибута E-mail не начинаются на определенное буквосочетание. Для вариантов 5,9,13 выбрать фирмы, где значения расчетного счета не начинаются на определенные цифры. Для вариантов 11,14 выбрать фирмы, где значения даты годности или выполнения заказа соответственно не начинаются на определенные цифры. Для вариантов 1,10 выбрать города или сотрудников, названия (имена) которых не начинаются на определенную букву.

2. Самостоятельно получить все заказы определенной фирмы (одна из существующих в вашей БД). (к запросу 3)

3. Самостоятельно получить наименования товаров, купленных фирмой с кодом 3. (к запросу 4)

4. Для четных вариантов определить количество полученного товара (всего по Получено), для нечетных – количество оставшегося (всего по Остатку).

Контрольные вопросы

1. Опишите технологию создания запроса средствами SQL.
2. Какую структуру имеет запрос на выборку средствами SQL?
3. Как выполнить сортировку записей в SQL запросе?
4. Что выполняет данный запрос:

```
SELECT Фамилия, оклад  
FROM Сотрудники  
WHERE Оклад>5000?
```

5. Что выполняет данный запрос?

```
SELECT Код_отдела, SUM(Количество)  
FROM Сотрудники  
GROUP BY Код_отдела?
```

ЛАБОРАТОРНАЯ РАБОТА №7

Разработка отчетов

Цель работы. Используя стандартные средства Access, разработать отчеты к базе данных.

Средства Access по разработке отчетов предназначены для создания макета отчета, по которому может быть осуществлен вывод данных из таблиц в виде выходного печатного документа.

При необходимости вывода в отчете данных из нескольких таблиц в качестве основы для отчета может быть использован многотабличный запрос.

Создавая отчет, можно группировать данные по нескольким уровням. Для каждого уровня может производиться вычисление итогов, определяться заголовки и примечания по каждой группировке.

Отчет можно создавать с помощью *мастера* или в режиме *конструктора отчетов*.

Контрольный пример

1. Создание отчета с помощью Мастера

1) В окне базы данных щелкнем по закладке *Отчет*, а затем нажмём на кнопку *Создать*.

2) В окне диалога *Создание отчета* выберем из списка *Мастер отчетов*, а ниже выберем таблицу (например, таблица **Фирма**), раскрыв предлагаемый список (рис. 33.), затем нажмём кнопку *OK*.

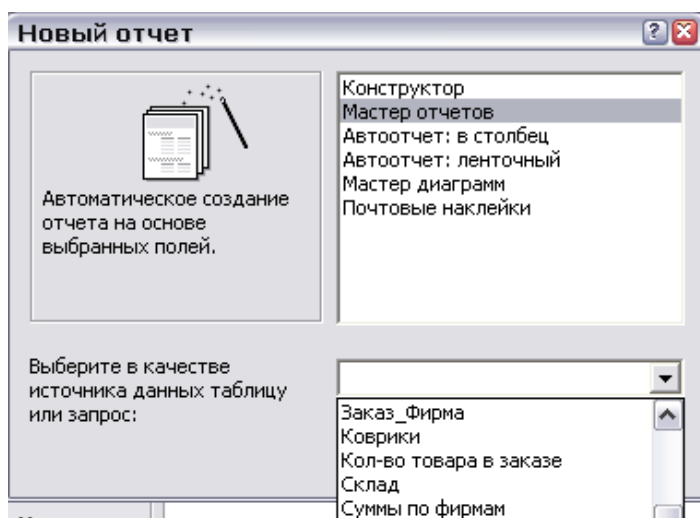


Рис. 33. Создание отчета в режиме мастера отчетов

3) В окне *Мастера по созданию отчетов* выберем поля для отчета (Код фирмы указывать не обязательно), с помощью кнопок переместим поля в правую часть окна. Нажмём кнопку *Далее* (рис. 34).



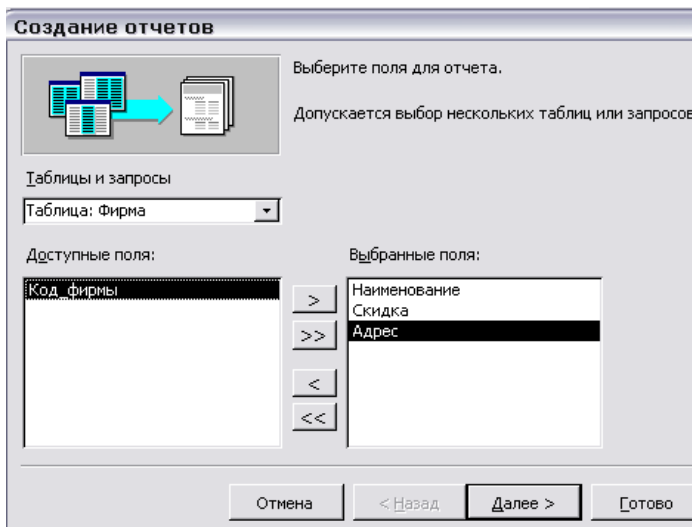


Рис. 34. Выбор полей для отчета

- 4) Выберем уровень группировки по Наименованию фирмы.
- 5) Порядок сортировки укажем – Отсутствует.
- 6) Выберем ступенчатый макет.
- 7) Выберем понравившийся стиль.
- 8) Укажем имя Отчета. Нажмём кнопку *Готово* (рис. 35).
- 9) Закроем документ отчета.

Фирмы		
Наименование	Скидка	Адрес
Абет	4	7-я линия, 49
Адвентисл и Ко	8	Ленина, 46
Апертские словари	5	11-я линия, 48
Вард и Ко	3	1-я 2
Гмагря+	6	8-я линия, 46
Игнат	8	Шиллинская 2

Рис. 35. Внешний вид отчета по фирмам

2. Создание отчета с помощью **Конструктора**

Отчеты можно создавать и вручную – в режиме *Конструктора*.

Для этого необходимо:

- 1) В окне базы данных выберем вкладку *Отчеты*.
- 2) Нажмём кнопку *Создать*.
- 3) В окне диалога *Новый отчет* выберем *Конструктор*.
- 4) Выберем имя таблицы или запроса, содержащих данные, по которым строится отчет. Для того чтобы использовать в отчете данные из нескольких таблиц, создавайте отчет на основе запроса.

5) После нажатия кнопки *ОК* новый отчет будет открыт в окне *Конструктора отчетов* (рис. 36).

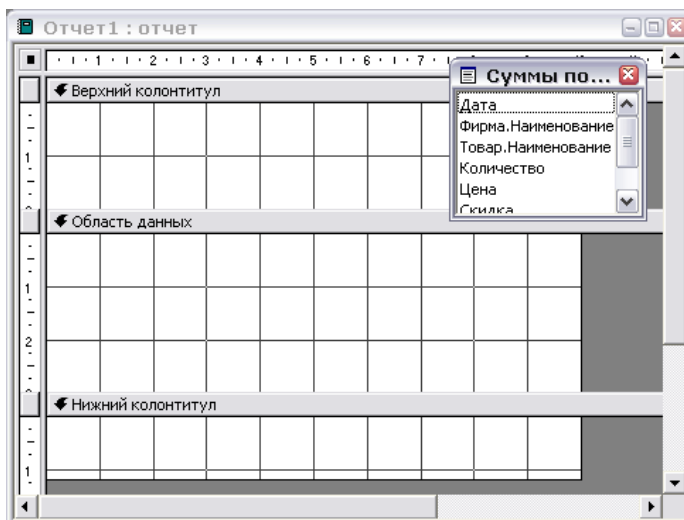


Рис. 36. Форма отчета

- 6) По умолчанию отчет состоит из 3-х частей:
 - верхний колонтитул. Здесь указывается информация об отчете, т.е.: название отчета, реквизиты, наименования полей и др.;
 - область данных. В этой части отчета выводятся данные из таблицы или запроса;
 - нижний колонтитул. Здесь указывается сводная информация об отчете, т.е. дата, количество страниц, др.

Редактирование отчета происходит так же, как и экранных форм. Создадим по вышеописанной схеме прайс-лист компании. 1) Выберем в качестве источника данных таблицу Товар. 2) Перетаскиваем на лист в область данных поля Наименование и Цена. 3) Подписи полей переносим в область верхнего колонтитула, там же указываем название отчета

и реквизиты фирмы. 4) В области нижнего колонтитула размещаем поля с информацией о текущей дате и номере страницы. Для этого в созданное поле вводится выражение с помощью *Построителя выражений* (контекстное меню ~ Свойства ~ закладка Данные ~ Данные (кнопка с [...])), где выбирается тип *Общие выражения*.

3. Отчеты с группировкой и вычислением итогов

Для сложных отчетов с группировкой и подведением итога оптимально создание отчета с помощью *Мастера* с доработкой в *Конструкторе*. Так, при создании отчета **Суммы по фирмам** необходимо сгруппировать проданные товары по заказам (по датам), а заказы – по фирмам-клиентам и вычислять суммы по каждому клиенту и в целом по обороту. Для этого необходимо при создании отчета *Мастером* выполнить следующие действия. 1) В качестве источника данных выбрать запрос **Сумма по фирмам 1**. 2) Поместить на отчет все поля запроса. 3) Указать уровни группировки сначала по названию фирмы, а потом по дате (рис. 37.) В окне *Интервалы группировки* (щелчок по кнопке *Группировка*) для даты поставить *обычный* интервал.

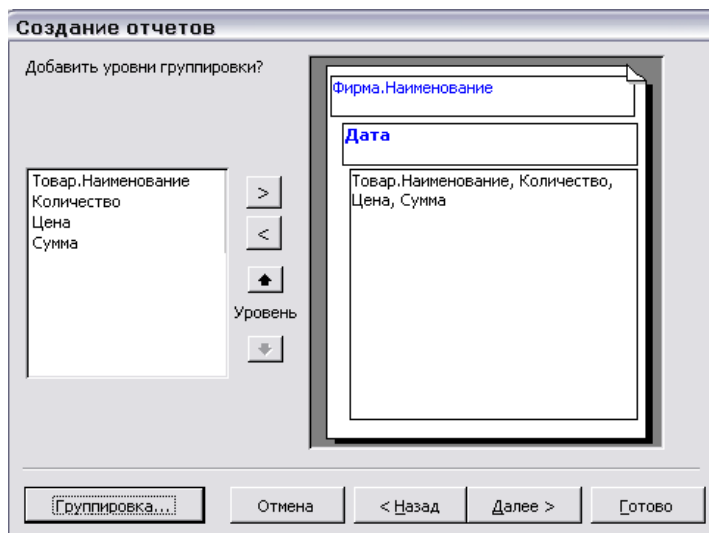


Рис. 37. Задание группировки в отчете

5) Для подведения итогов на следующем шаге создания отчета (окно сортировки) нужно нажать кнопку *Итоги* и в появившемся окне выбрать необходимые статистические формулы и поля, к которым они должны быть применены (рис. 38).

6) На следующих шагах мастера выбрать макет, стиль, имя отчета.

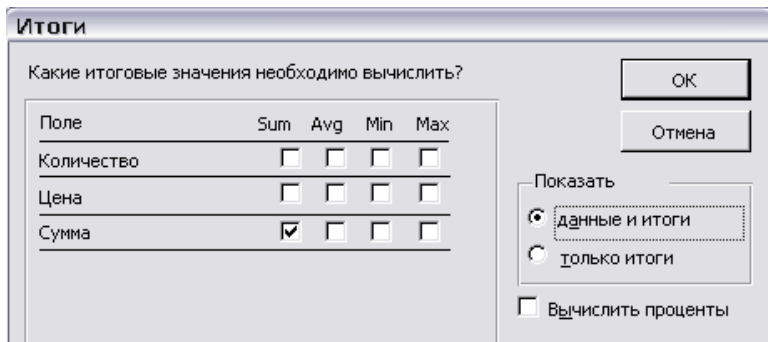


Рис. 38. Задание итоговых значений

7) После окончания работы *Мастера* отчет можно откорректировать в *Конструкторе*: убрать лишние итоговые значения (по умолчанию итоги считаются на каждом уровне группировки), отредактировать заголовки полей, оформление и пр.

Пример созданного отчета с применением *Мастера* и доработанного в режиме *Конструктора* представлен на рис. 39, а его вид в режиме *Предварительного просмотра* на рис. 40.

Суммы по фирмам					
Фирма.	Дата	Товар	Количество	Цена	Сумма
Фирма.Наименование					
Фирма.Наименование					
Дата					
=Format\$([Дата])					
Товар.Наимен.					
Товар.Наимен.					
Итого по фирме: =Sum([Сумма])					
="Страница " & [Page] & " из " & [Pages]					
Итого: =Sum([Сумма])					

Рис. 39. Вид отчета в режиме *Конструктора*

Суммы по фирмам					
Фирма.	Дата	Товар	Количество	Цена	Сумма
В ар я и Ко					
	12 Декабря 200	молоко	340	4,00р.	1319,2
	12 Декабря 200	табуретка	1	152,00р.	147,4594
		лезвие бритвы	5	28,00р.	135,8
		Итого по фирме:			1602,4594
Измаятков Инс.					
	15 Март 2002 г	коврик	2	3 000,00р.	5400
	23 Апрель 200	молоко	2	4,00р.	7,2

Рис. 40. Отчет о суммах при предварительном просмотре

Индивидуальное задание

1. Для нечетных вариантов создать отчет, о количествах товаров на складе. Для этого предварительно создать запрос по таблицам **Товар** и **Склад**, а отчет создавать на основании этого запроса.

2. Для четных вариантов требуется создать отчет о всех заказах, сделанных в определенный день. Для этого нужно создать запрос наподобие запроса **Суммы по фирмам**, но с указанием конкретной даты. Предварительно необходимо, чтобы хотя бы два заказа были сделаны в один и тот же день (таблица **Заказ_Фирма**, добавить записи).

Контрольные вопросы

1. Какие средства создания отчетов вы знаете?
2. Как осуществить расчет итогов в отчетах?
3. Как поместить в отчет данные из нескольких таблиц?
4. Какая информация помещается в область нижнего колонтитула при формировании отчета?
5. Какая информация помещается в область данных при формировании отчета?

ЛАБОРАТОРНАЯ РАБОТА № 8

Создание приложения пользователя

Цель работы. Для организации эффективной работы пользователя нужно создать целостное приложение данной предметной области, в которой все компоненты приложения должны быть сгруппированы по функциональному назначению.

Большое количество объектов БД, не сгруппированных по функциям приложения, затрудняет выполнение пользователем задач обработки данных в автоматизированной предметной области.

Для объединения объектов в едином диалоговом приложении создаются так называемые *кнопочные формы*. Кнопочная форма является панелью управления приложением. Кнопки такой формы обеспечивают вызов других кнопочных форм, а также отдельных объектов – отчетов, форм, макросов, модулей, с которых начинается решение задачи. Сюда же помещаются кнопки для возврата к кнопочным формам предыдущих уровней, выхода из Access.

В Access имеется средство автоматизированной разработки формы по управлению приложением, называемое *диспетчером кнопочных форм*.

Другим способом создания целостного приложения является организация пользовательской панели управления – *меню системы*. Ее пункты, в том числе и вложенные, выполняют те же функции, что и кнопки кнопочной формы.

Контрольный пример

Создание панели управления приложения

Как было указано выше, на панели управления приложением размещаются кнопки для вызова таблиц, форм и отчетов. Вызов же запросов происходит через запуск макросов.

1. Создание кнопочной формы.

1) Создадим макросы для вызова всех созданных запросов.

Для создания макросов перейдем на закладку *Макросы* и выберем кнопку *Создать*.

В окне *Макроса* выберем макрокоманду *Открыть Запрос* (рис. 41). В поле *Имя запроса* указываем имя соответствующего запроса, например **Товары по дате**.

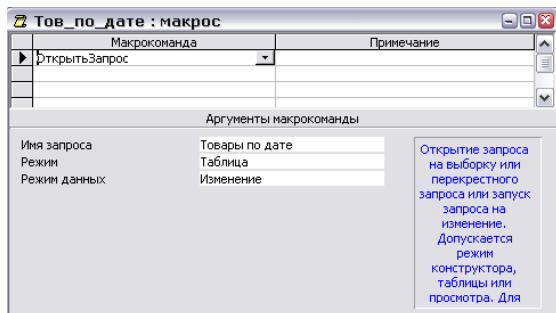


Рис. 41. Создание макроса для открытия запроса **Товар по дате**

Остается только закрыть запрос и дать ему имя (может совпадать с именем запроса). Повторите аналогичные действия для макросов всех запросов.

2) Создадим главную кнопочную форму приложения **Учет заказов**

Выберем пункт меню Сервис~Службные программы~Диспетчер кнопочных форм.

В окне Диспетчера кнопочных форм подтвердим создание кнопочной формы.

В следующем окне Диспетчера кнопочных форм появится окно со строкой **Главная кнопочная форма** (рис. 42). В этом окне диспетчера кнопочных форм формируется список кнопочных форм разных уровней.

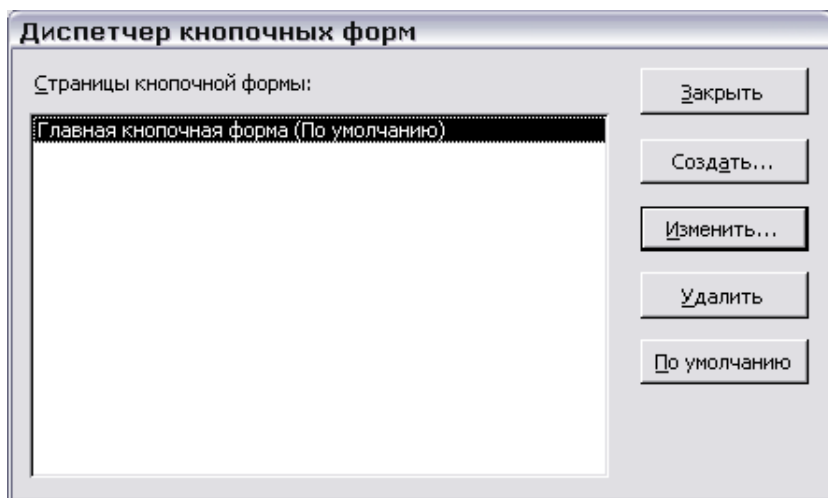


Рис. 42. Окно списка кнопочных форм приложения

3) Следующим шагом создадим другие кнопочные формы приложения:

Создадим три кнопочные формы для решения задач Ввода и редактирования данных, Поиска информации и Формирования отчетов.

Для создания кнопочной формы в окне диспетчера кнопочных форм нажмем кнопку *Создать*. В окне *Создание* в поле *Имя страницы кнопочной формы* (рис. 43) введем имя первой кнопочной формы **Ввод и редактирование данных**.

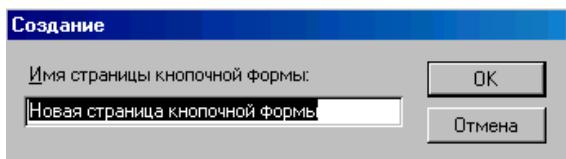


Рис. 43. Окно создания пустой кнопочной формы

После нажатия кнопки *Ок* в окне диспетчера кнопочных форм появится строка **Ввод и редактирование данных**.

Аналогичными действиями создаем кнопочные формы **Поиск данных** и **Отчетные формы**.

4) Отредактируем имена и сформируем элементы кнопочной формы.

Элементами кнопочных форм являются кнопки с подписями. Подпись задается пользователем и должна по возможности кратко и точно описывать действия, которые будут выполняться при нажатии кнопки. Для формирования элементов главной кнопочной формы в окне *Диспетчера кнопочных форм* выделим соответствующую строку и нажмем кнопку *Изменить*.

В окне диалога *Изменение страницы кнопочной формы* (рис. 44) переименуем **Главная кнопочная форма** в **Приложение «Учет заказов»**, введя это имя в поле *Название кнопочной формы*.

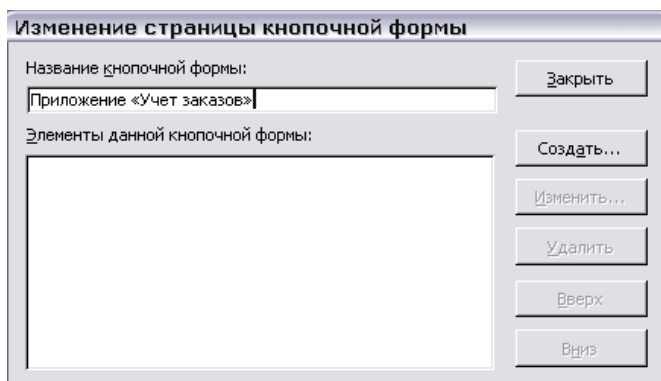


Рис. 44. Окно создания кнопочной формы и ее элементов

Элементы, создаваемые в кнопочной форме, могут быть подразделены на два типа:

- Элементы, предназначенные для организации вызова других кнопочных форм;
- Элементы, обеспечивающие такие действия как открытие формы, отчета, запуск макроса, программы и выход из приложения с закрытием базы данных.

4.1. Сформируем элементы для вызова кнопочных форм:

Для формирования в главной кнопочной форме **Приложение «Учет заказов»** кнопки вызова подчиненной формы нажмем кнопку *Создать*.

В окне *Изменение элемента кнопочной формы* в строке *Команда* выберем **Переход к кнопочной форме**. В строке *Кнопочная форма* выбирается форма, к которой должна обеспечивать переход создаваемая кнопка текущей формы. Откроем список созданных для приложения кнопочных форм и выберем **Ввод и редактирование данных**. В строке *Текст* введем подпись для этой кнопки **Ввод и редактирование данных**. Результат действий представлен на рис. 45. Завершим операцию нажатием кнопки ОК.

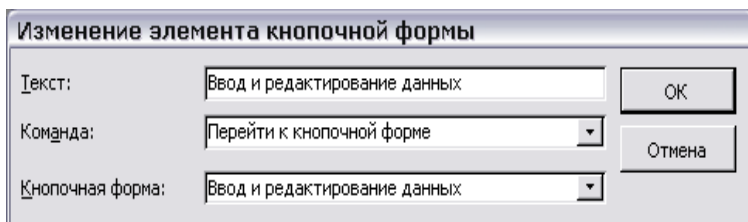


Рис. 45. Окно создания элемента кнопочной формы

Аналогичным образом создаем кнопки вызова других подчиненных форм. В поле элементы данных у вас должны появиться три созданных элемента.

По окончании этого процесса закрыть окно *Изменение страницы кнопочной формы* и перейти в окно *Диспетчера кнопочных форм*.

4.2. Формируем элементы для открытия объектов приложения:

Для формирования кнопок, обеспечивающих выполнение пользователем своих задач, необходимо, чтобы они были связаны с одним из объектов приложения. Для связи создаваемой кнопки с нужным объектом надо в окне *Изменения объекта кнопочной формы* выбрать в строке *Команда* одну из команд:

- Открытие формы в режиме добавления
- Открытие формы в режиме редактирования
- Открытие отчета

- Запуск макроса
- Запуск программы

Третья строка этого же окна будет содержать наименование формы, отчета или макроса, соответствующее выбранной во второй строке команде.

1. Создадим в кнопочной форме **Ввод и редактирование данных** кнопку для работы с формой **Товары**.

В окне *Диспетчера кнопочных форм* выберем форму **Ввод и редактирование данных**. Для создания элементов в этой форме нажмем кнопку *Изменить*. В открывшемся окне *Изменение страницы кнопочной формы* нажмем кнопку *Создать*.

В окне *Изменение элемента кнопочной формы* выберем команду **Открыть форму для изменения**. В появившейся строке *Форма* откроем список доступных форм и выберем форму **Товар**. В строку *Текст* введем подпись кнопки **Справочник Товаров**. Результат действий представлен на рис. 46.

Изменение элемента кнопочной формы	
Текст:	Справочник товаров
Команда:	Открыть форму для изменения
Форма:	товар
<input type="button" value="ОК"/> <input type="button" value="Отмена"/>	

Рис. 46. Окно создания элемента кнопочной формы

2. Аналогичным образом создаем кнопки для запуска остальных необходимых форм, запросов и отчетов, не забывая, что запросы открываются через макросы.

В каждой из подчиненных кнопочных форм необходимо создать кнопку для возврата к главной форме **Приложение «Учет заказов»**.

Для этого в окне *Изменение элемента кнопочной формы* для каждой из создаваемых форм (кроме формы по умолчанию) введем параметры, приведенные на рис. 47.

Изменение элемента кнопочной формы	
Текст:	На главную
Команда:	Перейти к кнопочной форме
Кнопочная форма:	Приложение «Учет заказов»
<input type="button" value="ОК"/> <input type="button" value="Отмена"/>	

Рис. 47. Создание кнопки для возврата в главную форму

После создания всех необходимых кнопок закрываем *Диспетчер кнопочных форм*. В списке форм базы данных появляется форма с именем **Кнопочная форма**. Внешний вид данной формы можно отредактировать в режиме *Конструктора*. Для запуска созданного приложения необходимо эту форму открыть. Внешний вид такой формы представлен на рис. 8. Если вы случайно удалите кнопочную форму из приложения, то настройки её останутся и решением в этом случае будет создание пустой кнопочной формы в новой базе данных и копирование её в исходное приложение. Все настройки будут перенесены на эту новую форму.

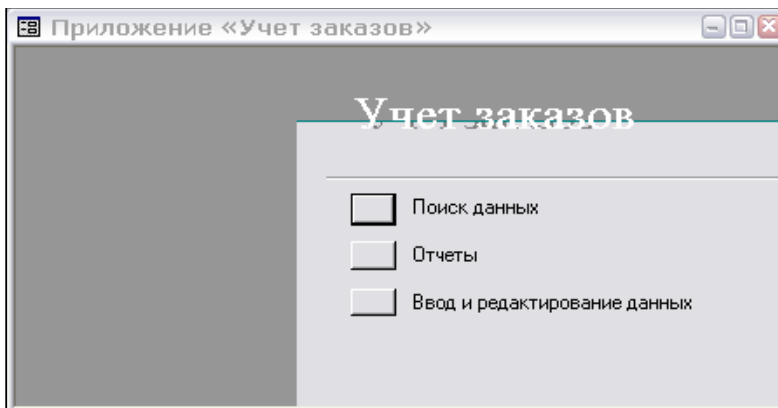


Рис. 48. Внешний вид главной страницы кнопочной формы

2. Создание панели пользовательского меню. Панель пользовательского меню является альтернативой кнопочной формы.

Воспользуемся пунктом меню *Вид ~ Панели инструментов ~ Настройка ~ вкладка Панели инструментов*. В появившемся окне нажмем кнопку *Создать*, назовем новую панель *Учет заказов*. Перейдем на вкладку *Команды* и добавим на панель путем перетаскивания из категории *Новое меню* команду *Новое меню*. Переименуем ее во *Ввод и редактирование данных*. Затем выберем категорию *Все формы* и из появившегося рядом списка форм перетащим в меню необходимые (рис. 49). Аналогично создадим выпадающие меню *Поиск* и *Отчетные формы*, перетащим туда созданные макросы (ИЛИ запросы), формы и отчеты. Закройте форму *Настройка*. Проверьте работу системы через вновь созданное меню.

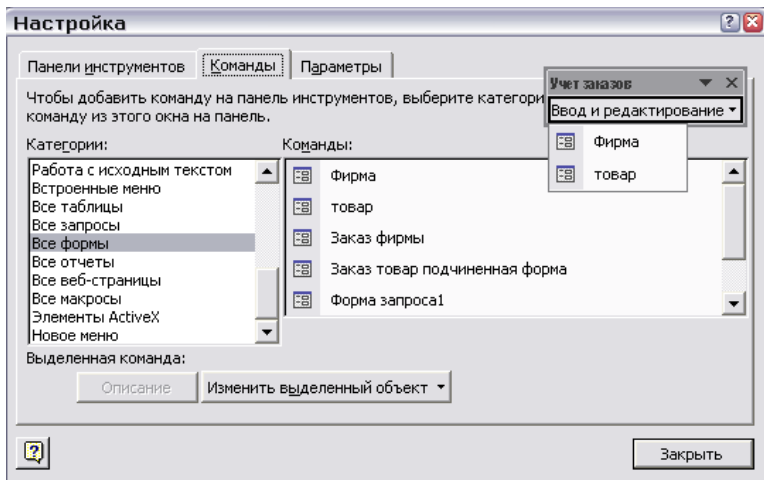


Рис. 49. Окно создания меню приложения

Индивидуальное задание

1. Для четных вариантов создать макросы ко всем запросам, построенным в режиме Конструктора (Л.р. №3), для нечетных – к запросам, написанным на языке SQL (Л.р. №6) Учесть при этом, что для запроса с произвольной выборкой необходимо открытие формы, а не запроса.

2. Откорректировать кнопочную форму в соответствии с появившимися элементами.

3. Вариантам 1,4,7,10,13 – организовать на кнопочной форме работу со страницами доступа (web-страница **Фирмы**), аналогично другим элементам БД. Вариантам 2,5,8,11,14 – реализовать возможность изменения приложения (кнопочной формы) из самого приложения. (добавить на форму определенную команду из стандартного списка). Вариантам 3,6,9,12,15 – добавить в меню системы команду, позволяющую автоматически конвертировать таблицы Access в таблицы Excel.

Контрольные вопросы

1. Опишите технологию создания макросов в Access.
2. С какой целью создается приложение пользователя?
3. Для чего служит диспетчер кнопочных форм?
4. Как создать страницу кнопочной формы?
5. Как создать меню пользователя в Access?

СПИСОК РЕКОМЕНДОВАННОЙ ЛИТЕРАТУРЫ

Основная литература

Дейт, К., Дж. Введение в системы баз данных. 6-е изд. – СПб.: «Вильямс», 2000. – 848 с.

Хомоненко А.Д., Цыганков В.М., Мальцев М.Г. Базы данных: Учебник для высших учебных заведений / Под ред. проф. А.Д. Хомоненко. – СПб.: КОРОНА принт, 2002. – 672 с.

В.В. Корнеев, А.Ф. Гареев, С.В. Васютин, В.В. Райх Базы данных. Интеллектуальная обработка информации. – М.: Нолидж, 2001. – 496 с.

Праг К., Ирвин М. Библия пользователя Access для Windows 2000. – К.: Диалектика, 2001 – 576 с.

Гарнаев А.С. Самоучитель VBA. – СПб., 1999. – 104 с.

Дополнительная литература

Алан Саймон Стратегические технологии БД. – М.: Финансы и статистика, 1999. – 484 с.

Хансен Г., Хансен Д. Базы данных. Разработка и управление. – М.: Бином, 2000. – 704 с.

Дж. Ульман, Дж. Видом. Введение в системы баз данных. – М.: Лори, 2000. – 374 с.

Ревунков Г.И., Самохвалов Э.Н., Чистов В.В. Базы и банки данных и знаний: Учебник для вузов по специальности АСУ. – М.: Высшая школа, 1992. – 367 с.

Мещеряков Е.В., Хомоненко А.Д. Публикация баз данных в Интернете. – СПб.: БХВ-Петербург, 2001. – 572 с.

Visual Basic для приложений (версия 5) в подлиннике П. Сканна и др – СПб.: «ВНУ-Санкт-Петербург», 1997. – 468 с.

Бекаревич Ю.Б., Пушкина Н.В. СУБД Access для Windows 95 в примерах. – СПб.: ВНУ-Санкт-Петербург, 1997. – 400 с.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1. ОРГАНИЗАЦИОННО-МЕТОДИЧЕСКИЕ УКАЗАНИЯ.....	4
1.1. Цель и задачи лабораторного практикума	4
1.2. Требования к знаниям и умениям, приобретаемым при выполнении лабораторного практикума	4
1.3. Объем и сроки изучения курса	5
1.4. Техническое и программное обеспечение для выполнения лабораторного практикума	5
1.5. Виды контроля знаний студентов и их отчетности по лабораторному практикуму.....	5
2. ЛАБОРАТОРНЫЙ ПРАКТИКУМ ПО КУРСУ БД.....	6
Общие указания к выполнению лабораторных работ	6
ЛАБОРАТОРНАЯ РАБОТА № 1.....	6
ЛАБОРАТОРНАЯ РАБОТА № 2.....	13
ЛАБОРАТОРНАЯ РАБОТА № 3.....	18
ЛАБОРАТОРНАЯ РАБОТА № 4.....	27
ЛАБОРАТОРНАЯ РАБОТА № 5.....	33
ЛАБОРАТОРНАЯ РАБОТА № 6.....	38
ЛАБОРАТОРНАЯ РАБОТА №7.....	40
ЛАБОРАТОРНАЯ РАБОТА № 8.....	47
СПИСОК РЕКОМЕНДОВАННОЙ ЛИТЕРАТУРЫ.....	54

Учебное издание

БАЗЫ ДАННЫХ

Практикум

**Составители: С.Л. Бедрина, О.Б. Богданова,
М.Л. Маевская**

В авторской редакции

Компьютерная верстка М.А. Портновой

Лицензия на издательскую деятельность ИД № 03816 от 22.01.2001

Подписано в печать 10.03.05. Формат 60×84/16.
Бумага писчая. Печать офсетная. Усл. печ. л. 3,25.
Уч.-изд. л. 3,2 Тираж 100 экз. Заказ

Издательство Владивостокского государственного университета
экономики и сервиса

690600, Владивосток, ул. Гоголя, 41
Отпечатано в типографии ВГУЭС
690600, Владивосток, ул. Державина, 57