

## ИННОВАЦИОННАЯ ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА



**Проект 2:** индивидуальная траектория обучения и качество образования

**Цель:** ориентированное на требования рынка образовательных услуг улучшение качества подготовки и переподготовки специалистов

Федеральное агентство по образованию

Государственное образовательное учреждение  
высшего профессионального образования

Владимирский государственный университет

Кафедра автоматизации технологических процессов

# МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ЛАБОРАТОРНЫМ РАБОТАМ ПО ДИСЦИПЛИНЕ «БАЗЫ ДАННЫХ»

Составитель  
А.Н. КИРИЛИНА

Владимир 2008

УДК 004.65  
ББК 32.97  
М54

Рецензент  
Доктор технических наук, профессор  
Владимирского государственного университета  
*И.Н. Егоров*

Печатается по решению редакционного совета  
Владимирского государственного университета

**Методические** указания к лабораторным работам по  
М54 дисциплине «Базы данных» / Владим. гос. ун-т ; сост.  
А. Н. Кирилина. – Владимир : Изд-во Владим. гос. ун-та,  
2008. – 24 с.

Позволяют в оперативные сроки освоить работу с системой управления базами данных (*Access*), содержат практические задания по разделам: создание таблиц и связей между ними, реализация запросов, создание экранных форм.

Предназначены для студентов очной формы обучения специальности 220301 – Автоматизация технологических процессов и производств (по отраслям), обучающихся по специализации «Компьютерные системы управления в производстве и бизнесе».

Ил. 2. Библиогр.: 7 назв.

УДК 004.65  
ББК 32.97

*Издание предназначено для изучения СУБД (систем управления базами данных) на примере программы Access. СУБД Access предназначена для разработки диалоговых офисных информационных систем, она использует реляционную модель данных и графический интерфейс Windows. СУБД Access входит в профессиональную версию офисной системы Microsoft Office.*

*Основная цель выполнения лабораторных работ – получить знания и укрепить навыки, достаточные для самостоятельного изучения СУБД.*

*Цикл содержит семь лабораторных работ с практическими заданиями по разделам: создание таблиц и связей между ними; реализация запросов; создание экранных форм.*

## **Лабораторная работа № 1. СОЗДАНИЕ ТАБЛИЦ**

**Цель работы:** получить понятие о шаблонах в *MS-Access*; изучить структуру объекта *MS-Access* «таблица»; научиться задавать полям различные типы данных.

### **Введение**

*MS-Access* является реляционной СУБД. Реляционная модель относится к классу структурированных моделей данных. Поэтому прежде чем данные могут быть введены в БД, структуру базы данных нужно спроектировать и описать. Принципы проектирования базы данных едины для всех реляционных СУБД.

СУБД *MS-Access* имеет следующую структуру:

- таблицы – для постоянного хранения данных;

- запросы – для вызова данных из таблиц по определенному шаблону и их дальнейшей обработки;
- формы – для удобного ввода и просмотра данных;
- отчеты – для удобного вывода заданной информации на печать;
- макросы – для задания свойств открываемых объектов и настройки порядка их работы;
- модули – для упрощения вызова ранее созданных функций.

Реляционная база данных представляет собой совокупность «плоских» таблиц. Таблицы состоят из полей и записей. Полями называются столбцы, а строки – записями. Внести запись в таблицу означает заполнить данными какую-нибудь строку. Чтобы создать таблицу необходимо определить ее поля, типы данных полей и, иногда, некоторые дополнительные свойства этих полей.

В *MS-Access* имеется несколько разных возможностей создания таблиц: в режиме конструктора, с помощью мастера и путем ввода данных.

Обычно наиболее простым путем создания каких-либо объектов является использование мастеров. Однако для таблиц наиболее используемым способом является их создание в режиме конструктора. Без понимания этого режима невозможно работать с БД. Кроме того, таблицы, созданные другими способами, часто требуется корректировать в режиме конструктора. Поэтому именно с этого режима следует начинать изучение работы с *MS-Access*.

### **Ход работы**

1. Запустите программу *MS-Access*.
2. В появившемся диалоговом окне установите флажок напротив опции «новая база данных» и подтвердите операцию.
3. В следующем диалоговом окне Вы должны выбрать папку для базы данных и задать ее имя (**Библиотека**).  
На экране появится окно с шестью вкладками, это и есть ваша база данных (она пока пустая).
4. Перейдите на вкладку «таблицы».

5. Щелкните мышью по кнопке «создать».

6. В появившемся диалоговом окне выберите режим создания (конструктор) и подтвердите данную операцию.

В столбец «имя поля» мы будем заносить имена столбцов нашей будущей таблицы (при этом нельзя использовать некоторые символы, в том числе точки и запяты). В столбце «тип данных» будем выбирать (используя кнопку вызова списка) тип данных. А то, что заносится в столбец «описание», затем появляется в виде комментариев в строке состояния (для проверки в одной из строк этого столбца напишите фразу: моя первая база данных).

Как видно из нижеперечисленных данных, нам необходимо создать следующие поля:

Название поля	Тип данных
Код книги	Счетчик
Автор	Текстовый
Название	Текстовый
Издательство	Текстовый
Год издания	Числовой
Количество страниц	Числовой
Тираж	Числовой
Статус издания	Текстовый

После ввода полей и типов данных желательно задать ключевое поле. Так как значения в ключевом поле должны быть уникальными, т.е. неповторяющимися, то в этом качестве следует выбрать поле **код\_книги**.

7. Для этого необходимо щелкнуть правой клавишей мыши по заданному полю и в появившемся меню выполнить команду ключевое поле.

8. Закройте окно конструктора и при запросе о сохранении задайте имя **Каталог**.

9. Откройте таблицу для заполнения данными. (Заметьте, что поле с типом данных счетчик будет заполнять автоматически).

## Данные для ввода

<b>Наименование издания</b>	<b>Статус издания</b>
1. Хомоненко, А.Д. Базы данных : учеб. для высш. учеб. заведений. – Изд. 3-е. – СПб. : КОРОНА принт, 2003. – 672 с. Тир. 3000 экз.	Учебник
2. Корнеев, В.В. Базы данных: интеллектуальная обработка информации : учеб. пособие / Рос. ассоциация изд. компьютерной лит. – М. : Нолидж, 2000. – 351 с. Тир. 2000 экз.	Учебное пособие
3. Вейскас, Дж. Эффективная работа с Microsoft Access 2000: рус. и англ. версии Access 2000: исчерпывающее руководство : пер. с англ. – СПб. : Питер, 2001. – 1036 с. : ил. + CD. Тир. 3000 экз.	Руководство пользователя
4. Грагер, Мартин. SQL : учеб. пособие : пер. с англ. – Изд. 2-е, перераб. и доп. – М.: Лори, 2003. – 643 с. : ил.	Учебное пособие
5. Кренке, Давид М. Теория и практика построения баз данных : учеб. пособие : пер. с англ. – 9-е изд. – СПб. : Питер, 2005. – 858 с. : ил. Тир. 1500 экз.	Учебное пособие
6. Новалис, С. Access 2000 : рук. пользователя VBA. – М. : Лори, 2001. – 50 с. : ил. Тир. 1000 экз.	Руководство пользователя
7. Праг, Керри Н. Access 2002. Библия пользователя : рук. пользователя : пер. с англ. / Керри Н. Праг, Майкл Р. Ирвин. – М.; СПб.; Киев: Диалектика, 2004. – 1207 с. : ил. +CD. Тир. 2000 экз.	Руководство пользователя
8. Питер, Роб. Системы БД: проектирование, реализация и управление : учебник : пер. с англ. / Питер Роб Карлос. – 5-е изд., доп. – СПб. : БХВ-Петербург, 2004. – 1024 с. :ил. Тир. 2000 экз.	Учебник
9. Ульман, Лари. MySQL : Рук. по изучению языка: пер. с англ. – М. : ДМК Пресс; СПб. : Питер, 2004. – 351 с. : ил. Тир. 3000 экз.	Руководство пользователя
10. Файли, Кри. SQL: рук. пользователя : пер. с англ. – М. : ДМК Пресс, СПб. : Питер, 2004. – 451 с. : ил. Тираж 4000 экз.	Руководство пользователя

10. Посчитайте максимальное количество символов в каждом из полей с текстовыми данными и, на всякий случай, прибавьте к значениям 5.

11. Откройте таблицу в режиме конструктора.

12. По очереди, переводя текстовый курсор в каждое из текстовых полей, задайте в окне свойств поля его размер (значения из п. 10).

13. Просмотрите, какие еще настройки можно задавать текстовым полям.

14. Просмотрите, какие настройки можно задавать другим полям. Закройте таблицу с сохранением.

## Лабораторная работа № 2. РЕЛЯЦИОННЫЕ БАЗЫ ДАННЫХ

**Цель работы:** получить понятие о реляционных базах и плюсах, предоставляемых ими.

### Введение

В процессе ввода данных очень часто возникает необходимость защитить оператора от ошибки и разграничить доступ к важной информации. Для этого данные разных категорий разделяют по разным таблицам, кроме этого, как правило, такое разделение позволяет более компактно хранить информацию. Данные в главной таблице индексируют и, используя эти индексы, связывают с подчиненными таблицами. Базы данных с такими связями называют реляционными.

### Ход работы

#### *Разделение данных на две таблицы*

1. Откройте базу данных **Библиотека**.
2. Создайте таблицу **Статус издания** с полями:

Название поля	Тип данных
Код_статуса	Счетчик
Статус издания	Текстовый

3. Поле **Код\_статуса** сделайте ключевым.
4. Создайте таблицу **Издательство** с полями:

Название поля	Тип данных
Код_издательства	Счетчик
Издательство	Текстовый

5. Поле **Код\_издательства** сделайте ключевым.
6. Заполните эти таблицы (таким образом, напротив каждого издательства и каждого статуса будет стоять его код).
7. Откройте таблицу **Библиотека** и замените названия в полях **Статус издания** и **Издательство** на соответствующие им коды в ранее созданных таблицах (для автоматизации попробуйте использовать команду замены из меню правка).
8. Откройте таблицу **Библиотека** в режиме конструктора.
9. Для полей **Статус издания** и **Издательство** измените тип данных на числовой.
10. Одновременно установив им размер поля **Длинное целое** (так как связываемые поля, как правило, должны иметь одинаковый тип данных).

*Установка связи между двумя таблицами*

1. Выполните команду **Схема данных** из меню **Сервис** (или нажмите на соответствующую пиктограмму на панели инструментов).
2. В диалоговом окне добавления таблиц добавьте в схему все три таблицы.
3. На поле **Код\_статуса** таблицы **Статус издания** нажмите левую клавишу мыши, и удерживая ее, перетащите на поле **Статус издания** таблицы **Библиотека**.

В появившемся диалоговом окне необходимо установить нужную связь: флажок напротив опции **обеспечение целостности данных** означает, что перед тем, как занести данные в подчиненную таблицу, программа будет проверять их на соответствие главной. (Таблица **Библиотека** является подчиненной для таблиц **Статус издания** и **Издательство**). Флажок напротив опции **каскадное обновление связанных полей** означает, что из-



менения в главной таблице автоматически будут влиять на подчиненную. Флажок напротив опции **каскадное удаление связанных полей** означает, что поля, удаленные в главной таблице, будут удалены и в подчиненной.

1. Установите все эти флажки.
2. Аналогичную операцию проделайте с таблицами **Библиотека** и **Издательство**.
3. Закройте схему с сохранением.
4. Попробуйте в таблице **Библиотека** в полях **Статус издания** и **Издательство** поменять коды на несуществующие в главных таблицах. Получилось? Почему?

### **Лабораторная работа № 3. СОЗДАНИЕ ЗАПРОСОВ НА ВЫБОРКУ**

**Цель работы:** изучить структуру баз данных **Запрос**; научиться пользоваться простейшими логическими операндами; получить практику выборки заданной информации.

#### **Введение**

Таблицы только хранят данные, но необходимо иметь возможность выбрать заданные данные из нескольких таблиц. Именно для этого служат **запросы на выборку**.

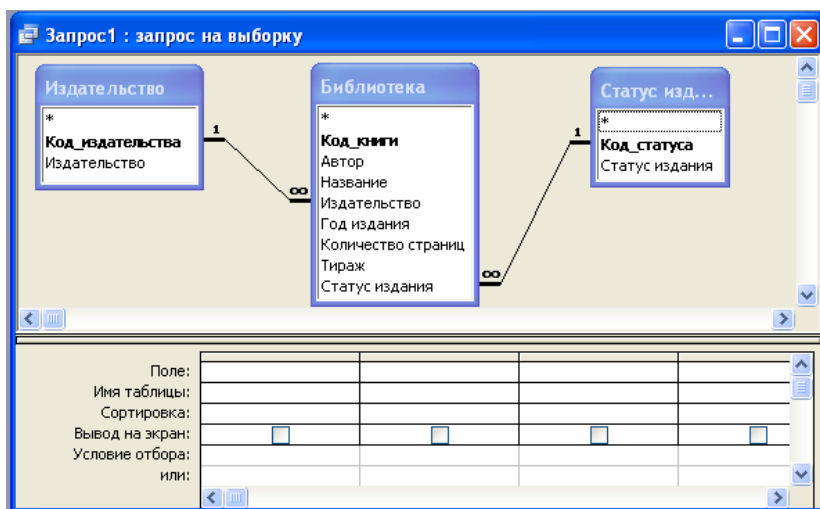
В **запросах на выборку** данные можно: отбирать по многим критериям; сортировать; с ними можно производить вычислительные операции.

Запрос – это временная таблица, данные в ней не хранятся постоянно, а только временно вызываются по заранее заданному шаблону в момент активизации запроса. Таким образом, в базе данных постоянно хранятся только шаблоны вызова данных (временные таблицы удаляются после закрытия запроса), а сама информация не дублируется.

Во время создания в запросе необходимо оставить ссылки на таблицы, откуда будут вызываться данные (добавление таблиц в запрос). Если нужная таблица не добавлена, то перед открытием запроса требуемые данные будут запрашиваться в диалоговом окне.

## Ход работы

1. Откройте базу данных **Библиотека** и перейдите на вкладку **Запросы**.
2. Создайте запрос, выбрав режим конструктора (появится окно конструктора запроса с диалоговым окном добавления таблиц. Окно добавления таблиц можно вызвать командой **Добавить таблицу** из меню **Запрос**).
3. Добавьте в запрос все три таблицы.
4. Убедитесь, что между добавленными таблицами автоматически установилась связь (см. рисунок).



Теперь необходимо выбрать поля для запроса. Из таблицы **Библиотека** возьмем поля: **Автор**, **Название**, **Издательство**, **Год издания**, **Количество страниц**, **Тираж**. Из таблицы **Статус издательства** поле **Статус издания**, а из таблицы **Издательство** поле **Издательство**. Это вызвано тем, что последние два поля в таблице **Библиотека** хранятся в кодовом виде, а в запросе мы можем позволить себе удобный просмотр.

5. Первое поле добавьте, произведя двойной щелчок по нему в таблице.

6. Второе перетащите, используя мышь (есть и третий способ, но он менее удобен).
7. Остальные поля добавьте любым способом.
8. Закройте запрос, при закрытии задав ему имя **Основной**.
9. Откройте запрос и просмотрите, какие данные он выводит, и закройте запрос.

Теперь наша задача настроить запрос на различные условия выборки. Для проверки выполненной работы создадим несколько запросов (на каждое условие по запросу). Чтобы не создавать заново почти одинаковые запросы, удобно несколько раз скопировать **Основной** через буфер обмена.

Для того чтобы вывести в запросе только книги определенного издательства, необходимо в поле запроса **Издательство** в строку **Условие отбора** ввести название издательства. Аналогично для любого поля.

Для того, чтобы запретить выводить в запросе книги определенного издательства, необходимо в данном поле и уже указанной строке ввести заданное название издательства, но перед ним поставить оператор *Not*.

Для того чтобы вывести книги с тиражом больше 2000, достаточно в поле **Тираж** в строку **Условие отбора** ввести  $>2000$ . Кстати, для условий предусмотрено две строки, поэтому для одного поля можно вводить два условия.

Можно сделать так, чтобы перед выводом запроса программа просила ввести данные, строки с которыми будут выведены. Для этого в нужном поле в указанной строке надо ввести [Введите данные для отбора]. Выражение в скобках может быть любым, но именно оно появится в диалоговом окне.

Для того чтобы находить в поле не конкретное значение, а только его фрагмент, используют оператор *Like*. Его ставят впереди заданного фрагмента; до или после фрагмента, в этом случае можно использовать звездочки маски.

10. Выделите запрос **Основной** (он должен быть закрыт).
11. Скопируйте его в буфер обмена.

12. Вставьте его в базу шесть раз, задавая имена, созвучные следующим ниже заданиям.
13. Создайте запрос, выводящий книги издательства СПб.: Питер.
14. Создайте запрос, не выводящий книги издательства СПб.: Питер.
15. Создайте запрос, выводящий книги с тиражом более 2000 и менее 3500.
16. Создайте запрос, выводящий книги тиражом более 3000 и количеством страниц менее 400.
17. Создайте запрос, спрашивающий: книги какого года издания вывести на экран?
18. Создайте запрос, выводящий названия книг авторов, чья фамилия начинается на букву Х.

#### **Лабораторная работа № 4. СОЗДАНИЕ ЗАПРОСОВ С ВЫЧИСЛЯЕМЫМИ ПОЛЯМИ**

**Цель работы:** практика создания баз данных; изучение построения математического выражения в структуре запрос; изучение групповых операций в запросах; практика работы с контекстным меню.

#### **Введение**

Как известно, базы данных нужны не только для хранения, но и для обработки данных. Одним из способов является создание вычисляемых полей в запросах на выборку. На примере этой лабораторной работе вы увидите, что этого способа иногда бывает достаточно для создания базы данных, имеющей практическое значение. В данной работе опущены многие подсказки по предыдущим темам. Вам самим предстоит решать, какие поля делать ключевыми и т.п.

## Ход работы

### Часть 1. Подготовка

1. Создайте базу данных Магазин
2. Создайте таблицу Товар по примеру:

Код товара	Товар	Цена поставки \$
1	Монитор	150,00
2	Винчестер	120,00
3	CD-ROM	70,00
4	Конверт для CD	1,00

Счетчик

Текстовый

Числовой

3. Создайте таблицу Продажа товара по дате по приведенному ниже примеру:

Код товара	Количество	Наценка	Дата
1	5	0,2	04.11.07
1	5	0,2	05.11.07
2	2	0,2	04.11.07
3	10	0,2	04.11.07
4	2	0,02	02.11.07

Числовой

Числовой

Числовой

Дата/время

- Примечания:* 1. После задания типа данных для поля **наценка** переведите текстовый курсор в поле **Размер поля**, и с помощью **кнопки выпадающего списка** выберите режим **Двойное с плавающей точкой (4 байта)**.
2. После задания типа данных для поля **дата** переведите текстовый курсор в поле **Формат поля** и с помощью **кнопки выпадающего списка** выберите режим **Краткий формат даты**.

4. Создайте таблицу Курс доллара по примеру:

Дата	Курс_\$
04.11.07	25
05.11.07	26
06.11.07	27
02.11.07	24
03.11.07	0

Дата/время

Денежный

5. Используя команду **Схема данных** из меню **Сервис**, свяжите таблицы **Товар** и **Продажа товара по дате** по полю код товара.
6. Создайте в режиме **Конструктора** запрос **прибыль по дате за вид товара** с полями товар из таблицы **Товар** и дата из таблицы **Курс доллара**.
7. **ВНИМАНИЕ!** Убедитесь, что в запрос вставлены сами эти таблицы.
8. Кроме того, для дальнейшей работы необходимо, чтобы в этот запрос была вставлена таблица **Продажа товара по дате**.
9. Установите (перетаскиванием мыши) связь между полями **Дата** таблиц **Продажа товара по дате** и **Курс доллара**. Сохраните запрос под указанным именем.

### *Часть 2. Построение вычисляемого поля*

Наша задача построить в следующем столбце данного запроса выражение для учета прибыли по виду товара за определенную дату в рублях. Для этого необходимо умножить столбец цен в долларах на курс доллара по датам, далее умножить получившееся произведение на столбец коэффициентов наценки и умножить на количество проданного товара по дате.

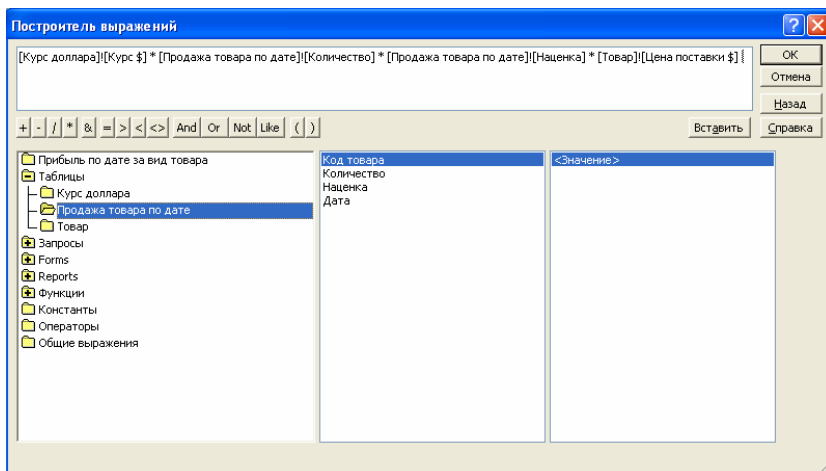
#### **Решение задачи:**

1. Откройте запрос **Прибыль по дате за вид товара** в режиме конструктора.
2. Установите текстовый курсор в строку Поле свободного столбца, и щелчком правой клавиши мыши на нем вызовите контекстное меню.
3. Выполните команду **Построить**.

*Примечание: появится диалоговое окно Построитель выражений (см. рисунок).*

4. В данном диалоговом окне, открывая папки Таблицы и Запросы, а затем заданные таблицы и запросы, требуется кнопкой вставлять в окно нужные поля, а также знаки необходимых математических операций.
5. Подготовив выражение, как указано в описании ко второй части, нажать подтверждающую клавишу.

- Установить флажок (щелчком мыши) в поле Вывод на экран создаваемого столбца.



- Сохраните измененный запрос (Файл – сохранить).
- Переведите текстовый курсор в начало строки Поле созданного столбца.
- Удалите название поля (Выражение1), сохранив двоеточие.
- Введите заданное название поля: **Прибыль по дате за вид товара**.
- В меню Вид выполните команду Режим таблицы.
- Если построенное поле выводит данные, закройте запрос с сохранением, иначе закройте без сохранения.

Мы добились учета прибыли магазина по виду товара за определенную дату. Теперь необходимо учесть полную прибыль магазина по дате. Для этого применяется функция группировки в создаваемом запросе. В такой запрос обычно вставляют поле или поля, по которым будет производиться группировка, и поле, которое по этим группировкам необходимо учитывать. В нашем случае группировать мы будем по полю **Дата**, а учитывать (применяя функцию суммы) поле, построенное в предыдущем запросе.

### *Решение задачи*

1. Создать запрос, добавив в него только предыдущий запрос (из него будем брать данные).
2. Вставить из запроса поля **Дата** и поле, построенное в предыдущем запросе.
3. Выбрать на панели инструментов или в контекстном меню операцию группировки (в запросе должна добавиться строка **Групповая операция**).
4. Установить текстовый курсор в ячейку на пересечении этой строки и столбца, построенного в предыдущем запросе.
5. Щелкнуть в строке **Групповая операция** по кнопке выпадающего списка и выбрать из него функцию **sum**.
6. Закрыть запрос с сохранением. Сохраните базу данных на дискету.

## **Лабораторная работа № 5. СОЗДАНИЕ И КОНСТРУИРОВАНИЕ ФОРМ**

**Цель работы:** знакомство со структурой формы; практика создания форм; практика конструирования форм; знакомство с режимом мастера.

### **Введение**

Форма – это структура базы данных, позволяющая удобно вводить и обрабатывать заданную информацию. Очень часто для компактного хранения информации требуется заносить в таблицу код объекта, а не его имя. Таким примером может служить таблица **Продажа товара по дате** из предыдущей лабораторной работы. Заполнение таких таблиц, особенно для рядовых операторов, было бы связано с большими проблемами. Формы позволяют, выбирая из списка имена объектов, вставлять в таблицы их коды. Все возможности форм нельзя рассмотреть в данном издании. В нем мы познакомимся лишь с некоторыми из них.



## Ход работы

### Часть 1. Создание формы

Формы и отчеты довольно сложные структуры баз данных, поэтому обычно их создают в режиме **Мастера** и дорабатывают в режиме **конструктора**. Так поступим и мы.

1. Откройте базу данных **Магазин** и перейдите на вкладку **формы**.
2. Щелкните по кнопке создать и выберите режим **мастер форм**.
3. В поле со списком **Таблицы/запросы** выберите таблицу-источник **Продажа товара по дате**.
4. Используя клавишу «>>», перенесите все поля данной таблицы из окна «доступные поля» в окно «выбранные поля» и перейдите к следующему диалоговому окну.
5. В нем установите флажок напротив опции «в один столбец» и перейдите к следующему диалоговому окну.
6. Выберите в нем оформление и перейдите к следующему диалоговому окну, где задайте имя форме и нажмите на кнопку **Готово**.

Вы создали форму, которая позволяет чуть более удобно вводить данные в таблицу **Продажа товара по дате**. Но для ее заполнения необходимо помнить коды всех товаров. Поэтому наша следующая задача обеспечить удобный ввод данных в поле **код товара** таблицы **Продажа товара по дате**.

### Часть 2. Доработка формы

Мы должны обеспечить, чтобы при вводе названия товара в нашей форме в таблицу заносился его код.

1. Откройте Вашу форму в режиме конструктора.
2. Выделите поле **код товара** и удалите его, используя клавишу **Delete**.
3. Найдите в окне программы панель элементов и перепишите возможные для вставки объекты.
4. Из элементов (щелчком мыши) выберите **поле со списком**.
5. Подведите курсор мыши к предполагаемому месту вставки нового поля (учтите, что слева от него автоматически вставится заголовок).

6. Нажмите левую клавишу мыши, и удерживая ее, растяните объект как рисунок (после того как Вы отпустите ее, у Вас появится первое окно диалога).
7. Ознакомьтесь с ним и установите флажок напротив нужной опции (первая).
8. Нажав клавишу **Далее**, перейдите к следующему диалоговому окну. В нем выберите таблицу, откуда будет браться наименование товара.
9. В следующем диалоговом окне выберите два поля (из одного будем брать коды товаров, а благодаря другому Вы сумеете видеть при этом названия товаров). Подсказка: эти поля – код товара и товар.
10. Со следующим диалоговым окном просто ознакомьтесь.
11. Далее выберите опцию **Сохранить в поле** и, используя кнопку вызова списка, выберите из него **код товара**.
12. В последнем окне задайте подпись созданного поля (**товар**). Закройте форму с сохранением.
13. Откройте форму и произведите ввод новой записи.
14. Проверьте, что попало в таблицу **Продажа товара по дате** в поле **код товара**.

## **Лабораторная работа № 6. СВОЙСТВА ФОРМ**

**Цель работы:** знакомство со структурой **форма**; практика создания форм; практика конструирования форм; знакомство с режимом **мастера**.

### **Введение**

У созданной формы в предыдущей лабораторной работе есть ряд недостатков: желая добавить новые записи, Вы можете испортить предыдущие; порядок ввода полей записи нарушен созданием нового поля; шрифт мелок и невыразителен. Устранить данные недостатки позволит задание неких параметров (свойств).

## Ход работы

### *Часть 1. Задание свойств формы*

1. Сделайте две копии своей формы. Присвойте полученным формам имена: **ввод новых данных**, **удаление записей** и **редактирование данных**.
2. Откройте форму **ввод новых данных** в режиме конструктора.
3. Щелкните правой клавишей мыши на заголовке окна формы.
4. Выберите из контекстного меню команду **свойства**.
5. В появившемся диалоговом окне перейдите на вкладку **все**.
6. Переведите текстовый курсор в строку **разрешить изменение** и, используя кнопку вызова списка, выберите **нет**.
7. Аналогичную операцию проделайте в строке **разрешить удаление**.
8. Закройте форму с сохранением, откройте ее и попробуйте:  
а) изменить запись; б) удалить запись; в) добавить запись.
9. Подумайте: какие изменения надо внести в две другие формы. Сделайте их.
10. Откройте одну из форм в режиме конструктора и вызовите на заголовке ее окна контекстное меню.
11. Выполните команду **последовательность перехода**.
12. В появившемся диалоговом окне измените последовательность так, чтобы поля шли как они расположены внутри формы. Проверьте сделанные изменения.
13. Откройте любую форму в режиме конструктора.
14. Вызовите контекстное меню на одном из полей формы.
15. Выполните команду свойства и в появившемся окне попробуйте самостоятельно изменить величину шрифта.

## Лабораторная работа № 7. СОЗДАНИЕ КНОПОЧНЫХ ФОРМ

**Цель работы:** получить практические навыки по созданию таблиц, представление о создании оболочек для баз данных.

### Введение

После создания реальной базы данных грамотно пользоваться ею может только создатель. Для её эксплуатации рядовым пользователем необходимо создать оболочку управления с понятным интерфейсом. Для этой цели подходят так называемые кнопочные формы. Идея в том, чтобы установить связь между определенным событием и щелчком мыши по графическому изображению кнопки. Однако не все события можно вызвать с использованием кнопочных форм, реализованных в СУБД *MS-Access*. Поэтому для обхода этого препятствия мы воспользуемся созданием макросов, которые содержат макрокоманды.

### *Часть 1. Подготовка*

1. Создайте базу данных – **Адресная книга**.
2. Создайте таблицу – **Адресат** с полями: код\_адресата, ФИО, страна, город, телефон, e-mail, адрес, примечание.
3. Создайте форму для заполнения данной таблицы.
4. Создайте запрос – поиск по фамилии, содержащий все поля данной таблицы за исключением поля код\_адресата.
5. Задайте в этом запросе условие отбора по вводимой фамилии (на пересечении столбца ФИО и строки условие отбора набрать [Введите ФИО]).

### *Часть 2. Создание кнопочной формы*

1. В меню **сервис** из списка **Служебные программы** выберите команду **Диспетчер кнопочных форм**.
2. Подтвердите ее создание (на экране появилось диалоговое окно, в котором высветилось название кнопочной формы, данное ей по умолчанию).
3. Для изменения данной кнопочной формы щелкните по кнопке **изменить**.

4. В следующем диалоговом окне в верхнем поле замените название кнопочной формы на название базы данных (у Вас есть кнопочная форма, но нет кнопок), щелкните по кнопке **создать**.
5. В появившемся окне (изменение элемента кнопочной формы) выберите, используя кнопки выпадающего списка: заданную команду (**открытие формы в режиме редактирования**); форму, которую следует открывать (она у вас единственная), и задайте (в верхнем поле) имя создаваемой кнопки.
6. Щелкните еще раз по кнопке **создать**. Убедитесь, что в поле команда нет команды открытия запроса. Нажмите клавишу отмены.
7. Закройте окна кнопочной формы. Перейдите на вкладку **Формы**.
8. Откройте кнопочную форму. Убедитесь, что созданная вами кнопка работает. Закройте кнопочную форму.

#### *Часть 3. Создание макроса, открывающего запрос*

1. Перейдите на вкладку **Макросы**. Выполните команду **Создать**.
2. В появившемся диалоговом окне в поле **Макрокоманда**, используя кнопку вызова списка, выберите команду **Открыть запрос**.
3. В поле **Аргументы** макрокоманды выберите имя открываемого запроса, его режим (таблица) и режим данных (чтение).
4. Закройте макрос, сохранив его под именем **Вызов**.

#### *Часть 4. Создание кнопочной команды, запускающей макрос*

1. Выполните команду **Диспетчер кнопочных форм** из списка служебных программ меню сервис.
2. В появившемся диалоговом окне нажмите кнопку **изменить**.
3. В следующем окне выполните команду создать.
4. Выполните настройки аналогично пунктам из ч. 2.

## РЕКОМЕНДАТЕЛЬНЫЙ БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Хомоненко, А. Д. Microsoft Access : быстрый старт / А. Д. Хомоненко, В. В. Гриндин. – СПб. : БХВ-Петербург, 2003. – 120 с. – ISBN 5-94157-195-X.

2. Диго, С. М. Базы данных: проектирование и использование : учеб. для вузов / С. М. Диго. – М. : Финансы и статистика, 2005. – 591 с. – ISBN 5-279-02571-2.

3. Чекалов, А. П. Базы данных: от проектирования до разработки приложений / А. П. Чекалов. – СПб. : БХВ-Петербург, 2003. – 380 с. – ISBN 5-94157-283-2.

4. Харитонов, И. Самоучитель ACCESS 2000 / И. Харитонов. – СПб.; М.; Харьков; Минск : ПИТЕР, 2001. – 383 с. – ISBN 5-272-00282-2 : 83-00.

5. Карпов, Б. MICROSOFT ACCESS 2000 : справочник / Б. Карпов. – СПб.; М.; Харьков; Минск : ПИТЕР, 2001. – 410 с. – ISBN 5-8046-0059.

6. Виллариал, Б. Программирование в Access 2002 : в примерах : пер. с англ. / Б. Виллариал. – М. : Кудиц-Образ, 2003. – 476 с. – ISBN 5-93378-059-6.

7. Золотова, С. И. Практикум по Access : подготовительный курс, предваряющий более глубокое изучение технологии баз данных / С. И. Золотова. – М. : Финансы и статистика, 2004. – 143 с. – ISBN 5-279-02284-5.

## ОГЛАВЛЕНИЕ

Лабораторная работа № 1 <b>Создание таблиц</b> .....	3
Лабораторная работа № 2 <b>Реляционные базы данных</b> .....	7
Лабораторная работа № 3 <b>Создание запросов на выборку</b> .....	9
Лабораторная работа № 4 <b>Создание запросов с вычисляемыми полями</b> .....	12
Лабораторная работа № 5 <b>Создание и конструирование форм</b> .....	16
Лабораторная работа № 6 <b>Свойства форм</b> .....	18
Лабораторная работа № 7 <b>Создание кнопочных форм</b> .....	20
<b>Рекомендательный библиографический список</b> .....	22

МЕТОДИЧЕСКИЕ УКАЗАНИЯ  
К ЛАБОРАТОРНЫМ РАБОТАМ  
ПО ДИСЦИПЛИНЕ «БАЗЫ ДАННЫХ»

Составитель  
КИРИЛИНА Анастасия Николаевна

Ответственный за выпуск – зав. кафедрой профессор В.Ф. Коростелев

Подписано в печать 20.06.08.  
Формат 60x84/16 Усл. печ. л. 1,39 Тираж 100 экз.  
Заказ  
Издательство  
Владимирского государственного университета.  
600000, Владимир, ул. Горького, 87.