

Министерство образования Республики Беларусь  
Учреждение образования  
«Белорусский государственный университет  
информатики и радиоэлектроники»

Кафедра экономической информатики

**А. А. Бутов**

***ТЕХНОЛОГИИ ОРГАНИЗАЦИИ, ХРАНЕНИЯ И ОБРАБОТКИ ДАННЫХ***

Лабораторный практикум  
для студентов экономических специальностей БГУИР  
всех форм обучения

Минск 2008

УДК 004.6 (076)  
ББК 32.973.26-018 я73  
Б 93

**Р е ц е н з е н т**  
зав. кафедрой ИИТ БГУИР, д-р техн. наук  
проф. В. В. Голенков

**Бутов, А. А.**

Б 93 Технологии организации, хранения и обработки данных : лаб. практикум для студ. экон. спец. БГУИР всех форм обуч. / А. А. Бутов. – Минск : БГУИР, 2008. – 52 с. : ил.  
ISBN 978-985-488-266-6

Представлен курс из шести лабораторных работ, предназначенных для практического освоения методов работы с базами данных в среде СУБД Microsoft Access. К каждой работе даны краткие теоретические сведения, необходимые для ее выполнения, и приведены варианты заданий.

**УДК 004.6 (076)**  
**ББК 32.973.26-018 я73**

**ISBN 978-985-488-266-6**

© Бутов А. А., 2008  
© УО «Белорусский государственный университет информатики и радиоэлектроники», 2008

## СОДЕРЖАНИЕ

<b>ВВЕДЕНИЕ .....</b>	<b>4</b>
<b>ЛАБОРАТОРНАЯ РАБОТА №1. СОЗДАНИЕ ТАБЛИЦ.....</b>	<b>6</b>
Методические указания.....	6
Задание к работе.....	9
<b>ЛАБОРАТОРНАЯ РАБОТА №2. СОЗДАНИЕ ЗАПРОСОВ .....</b>	<b>16</b>
Методические указания.....	16
Варианты заданий.....	20
<b>ЛАБОРАТОРНАЯ РАБОТА №3. СОЗДАНИЕ ФОРМ .....</b>	<b>25</b>
Методические указания.....	25
Варианты заданий.....	27
<b>ЛАБОРАТОРНАЯ РАБОТА №4. СОЗДАНИЕ ОТЧЕТОВ.....</b>	<b>34</b>
Методические указания.....	34
Варианты заданий.....	36
<b>ЛАБОРАТОРНАЯ РАБОТА №5. СОЗДАНИЕ МАКРОСОВ.....</b>	<b>42</b>
Методические указания.....	42
Варианты заданий.....	44
<b>ЛАБОРАТОРНАЯ РАБОТА №6. СОЗДАНИЕ ПРИЛОЖЕНИЙ .....</b>	<b>48</b>
Методические указания.....	48
Задание к работе.....	49
<b>ЛИТЕРАТУРА .....</b>	<b>50</b>

## Введение

В рамках дисциплины «Технологии организации, хранения и обработки данных» изучаются теоретические основы и практические методы проектирования баз данных. Настоящий практикум предназначен для освоения практических методов создания баз данных с помощью пакета прикладных программ Microsoft Access.

Microsoft Access представляет собой систему управления базами данных (СУБД), входящую в состав профессиональной редакции Microsoft Office и являющуюся одной из лучших в классе настольных СУБД. Область применения Access – от однопользовательских локальных разработок до несложных профессиональных сетевых проектов. Важнейшей особенностью этой СУБД является возможность разработки приложений для работы с базами данных без использования программного кода, лишь с помощью средств визуального проектирования, встроенных мастеров и шаблонов. При необходимости приложения можно усовершенствовать, используя встроенный язык Visual Basic for Application (VBA). Также реализована возможность использования языка структурированных запросов SQL. Дальнейшее наращивание функциональных возможностей связано с построением клиент-серверных систем, в которых Access выступает в качестве клиента, запрашивающего сервисы у мощной внешней СУБД Microsoft SQL Server, используемой в качестве сервера. Для реализации этой возможности используются проекты Microsoft Access (файлы с расширением .adp).

Важнейшей задачей любой СУБД, в том числе СУБД Access, является разработка баз данных, содержащих информацию, связанную с той или иной предметной областью. База данных, разрабатываемая в среде СУБД Access, может содержать в себе следующие основные объекты: таблицы, запросы, формы, отчеты, страницы, макросы и модули.

**Таблицы** непосредственно хранят информацию, относящуюся к конкретной предметной области.

**Запросы** позволяют получать всевозможные выборки информации из одной или нескольких таблиц и/или запросов с использованием различных критериев отбора данных.

**Формы** представляют собой окна с расположенными на них элементами управления (надписями, полями ввода, списками, выпадающими списками, переключателями, кнопками и др.) и реализуют графический интерфейс для взаимодействия пользователя с другими объектами базы данных, обычно таблицами и запросами.

**Отчеты** предназначены для описания формы и содержимого документов, выводимых на печать.

**Страницы** предназначены для публикации информации из базы данных на интернетовских страницах.

**Макросы** представляют собой последовательности команд на языке VBA, которые служат для автоматического выполнения некоторых операций или их

последовательностей, в частности, открытия запросов, печати отчетов, анализа данных в формах и отчетах, выдачи поясняющих сообщений и др.

**Модуль** представляет собой совокупность описаний и процедур на языке VBA, сохраненных под общим именем.

Все объекты базы данных тесно связаны между собой, причем важнейшими объектами являются таблицы, т. к. без них создание остальных объектов и, в первую очередь запросов, лишено какого бы то ни было практического смысла. В свою очередь, формы и отчеты, как правило, отображают данные из каких-либо таблиц и/или запросов, макросы же могут манипулировать всеми объектами базы данных и, прежде всего, формами и отчетами.

После запуска Access на экране появляется главное окно программы, которое содержит строку заголовка, строку главного меню, контекстно-зависимые панели инструментов, рабочее окно и строку состояния. Будет открыто также окно «Область задач», которое используется, например, для создания новой базы данных, просмотра и использования многоместного буфера обмена, выполнения поисковых операций в базе данных и др. Всегда можно воспользоваться справочной системой, позволяющей вызвать тот или иной раздел справочной информации.

# Лабораторная работа №1. Создание таблиц

## Цель работы

1. Ознакомиться с процессом создания таблиц в режиме конструктора.
2. Освоить способ установления связей между таблицами с помощью мастера подстановок.
3. Ознакомиться со схемой данных и настройкой параметров связей между таблицами.

## Задачи

1. Создание новой пустой базы данных.
2. Создание таблиц базы данных с заданными типами и свойствами полей.
3. Создание связей между таблицами с обеспечением для внешних ключей возможности подстановки в них значений из ключевых полей родительских таблиц.
4. Настройка связей между таблицами таким образом, чтобы обеспечить целостность данных.
5. Наполнение таблиц конкретными данными.

## Методические указания

Таблицы являются объектами базы данных, поэтому сначала необходимо создать собственно базу данных, в контексте которой таблицы и будут создаваться.

Существует два способа создания базы данных:

- 1) выбор типовой базы данных с помощью мастера;
- 2) создание пустой базы данных с последующим формированием в ней нужных объектов.

Первый способ обеспечивает быстрое создание одной из десяти типовых баз данных: «Контакты», «Мероприятия», «Проекты», «Расходы», «Основные фонды» и другие с образцами информации в таблицах. Если отсутствует подходящая база данных, следует выбрать ту, которая наиболее близка к решаемой задаче, и позже ее модифицировать.

Второй способ обеспечивает создание новой пустой базы данных.

В обоих случаях нужно указать имя создаваемой базы данных, которое будет дополнено расширением .mdb.

После создания базы данных на экране появится окно базы данных, в левой части которого расположены кнопки выбора типа объектов базы данных. Нажав одну из кнопок, мы увидим справа все имеющиеся объекты выбранного типа. Вверху расположены кнопки наиболее часто используемых режимов работы с объектами, а именно: **Открыть, Конструктор, Создать и Удалить.**

Для создания таблицы необходимо в окне базы данных выбрать слева кнопку **Таблицы**, а вверху кнопку **Создать**. Будет предложено пять вариантов создания таблицы:

- 1) режим таблицы;
- 2) мастер таблиц;
- 3) импорт таблиц;
- 4) связь с таблицами;
- 5) конструктор.

В режиме таблицы предлагается ввести названия полей и сами данные в пустую таблицу, при сохранении которых программа проводит анализ и автоматически определяет как число полей в таблице, так и приблизительный тип данных для каждого поля.

В режиме мастера таблиц предлагается пошаговое конструирование таблицы на основе полей, выбираемых из одной или нескольких типовых таблиц. Определяющим является выбор полей с нужным типом данных, а названия полей можно при необходимости изменить.

Импорт таблиц используется тогда, когда другая база данных содержит нужные нам таблицы. Импорт таблицы создает копию таблицы в разрабатываемой нами базе данных и никак далее не отражается на оригинале.

В отличие от импорта таблиц связывание только устанавливает связь с таблицами, хранящимися в другой базе данных, и поэтому возможна ситуация, когда таблицы будут изменены без нашего ведома.

Режим конструктора характеризуется наибольшей трудоемкостью, но и большей гибкостью и возможностью точно настроить свойства полей таблицы.

#### **Создание таблицы в режиме конструктора**

В режиме конструктора пользователь сам разрабатывает структуру создаваемой таблицы, а именно: задает имена полей и их типы, настраивает свойства каждого поля, а также формирует первичный ключ на основе одного или несколько полей.

Имеются следующие **типы полей** (указываются справа от имен полей в верхней части окна конструктора).

1. *Текстовый*. Для текстов и/или чисел, не требующих проведения расчетов. Размер поля – не более 255 символов.
2. *МЕМО*. Для очень длинных текстов. Размер поля – до 65 535 символов.
3. *Числовой*. Для чисел, используемых для проведения расчетов. Имеет целый ряд подтипов. Размер поля – от 1 до 16 байт.
4. *Дата/время*. Для даты и времени, относящихся к годам с 100 по 9999. Размер поля – 8 байт.
5. *Денежный*. Для денежных значений, используемых для проведения расчетов. Обеспечивает точность до 15 знаков в целой и до 4 знаков в дробной части. Размер поля – 8 байт.
6. *Счетчик*. Для уникальных последовательно возрастающих на единицу или случайных чисел, генерируемых автоматически. Размер поля – 4 байта.
7. *Логический*. Для значений «Да» и «Нет». Размер поля – 1 бит.

8. *Объект OLE* (Object Linking and Embedding). Для объектов, связанных или внедренных в таблицу Access (электронные таблицы Excel, документы Word, рисунки, звукозаписи и другие данные в двоичном формате). Размер поля – до 1 Гб.

9. *Гиперссылка*. Для комбинации текста и чисел, используемой в качестве адреса гиперссылки. Длина каждой из частей гиперссылки не более 2048 символов (отображаемый текст, адрес, дополнительный адрес, всплывающая подсказка).

Тип поля можно задать также при помощи мастера подстановок, который устанавливает, будет ли выбираться значение поля из другой, ранее созданной таблицы, или же из создаваемого списка значений. После завершения работы мастера устанавливается тип данных, основанный на типе подставляемых значений, а также устанавливается связь между таблицами, если значения будут выбираться из другой таблицы.

**Свойства полей** (указываются в нижней части окна конструктора, вкладка «Общие»)

1. *Размер поля*. Указывается для текстовых и числовых типов данных. Для текстовых полей изменяется в диапазоне от 1 до 255 символов, по умолчанию равен 50. Для числовых полей возможны 3 целочисленных подтипа (байт, целое, длинное целое), 3 вещественных подтипа (одинарное с плавающей точкой, двойное с плавающей точкой, действительное), а также подтип с названием «Код репликации».

2. *Формат поля*. Определяет вид представления информации при выводе на экран и на печать. Для числовых типов данных и типа Дата/время нужный формат можно выбрать из выпадающего списка или же ввести вручную.

3. *Число десятичных знаков*. Задаёт число десятичных знаков для полей вещественного и денежного типов.

4. *Маска ввода*. Используется при вводе данных, упрощая и проверяя правильность их ввода.

5. *Подпись*. Задаёт более подробное наименование поля, которое будет использоваться в формах и отчетах в качестве имени поля.

6. *Значение по умолчанию*. При формировании новой записи это значение будет подставляться в поле автоматически. При необходимости его можно изменить.

7. *Условие на значение*. Это выражение логического типа, которое используется для анализа вводимых в поле данных и блокирует их ввод, если указанное условие нарушается. Для формирования сложных условий можно использовать построитель выражений, который позволяет выбирать и вставлять в формируемое условие встроенные функции, константы, операторы сравнения, логические связки и др.

8. *Сообщение об ошибке*. Текст этого сообщения выдается на экран, если при вводе было нарушено рассмотренное выше условие на значение.

9. *Обязательное поле*. Здесь указывается необходимость ввода данных в поле для каждой записи таблицы.

10. *Индексированное поле.* Здесь указывается, нужно ли создавать индекс для поля. Индексы ускоряют выполнение запросов, а также операции сортировки и фильтрации записей по индексированным полям.

11. *Пустые строки.* Здесь указывается, допускается ли ввод в поле строк, не содержащих ни одного символа.

После создания таблиц нужно установить между ними связи (если только они не были созданы с помощью мастера подстановок). Обычно это связи типа «один ко многим», но иногда встречаются и «один к одному». Для создания связей необходимо открыть окно «Схема данных» и добавить в него все таблицы базы данных. Далее необходимо для каждой главной (родительской) таблицы определить подчиненную (дочернюю) таблицу и путем перетаскивания первичного ключа из главной таблицы на поле связи подчиненной таблицы создать между таблицами связь. При этом в появившемся окне «Изменение связей» для обеспечения целостности данных необходимо установить соответствующий флажок. При необходимости можно также установить флажки каскадного обновления и удаления данных.

После создания связей между таблицами можно переходить к вводу в них данных, начиная с главных таблиц, и только потом – подчиненных. Если главные таблицы заполнены, то ввод данных в поля связи (внешние ключи) подчиненных таблиц выполняется с помощью соответствующих выпадающих списков (если только типы полей связи были установлены при помощи мастера подстановок).

### **Задание к работе**

Создайте базу данных «Основные средства» (ОС), включающую в себя указанные ниже таблицы. Первое поле в каждой таблице сделайте первичным ключом. Типы данных, отмеченные звездочкой, устанавливайте с помощью мастера подстановок (строка запуска мастера подстановок находится в самом конце выпадающего списка с типами данных).

#### **Группы ОС**

<i>Наименование поля</i>	<i>Тип данных</i>	<i>Размер поля</i>	<i>Обязательное поле</i>
КодГруппыОС	Счетчик		
ГруппаОС	Текстовый	50	Да

#### **Виды ОС**

<i>Наименование поля</i>	<i>Тип данных</i>	<i>Размер поля</i>	<i>Обязательное поле</i>
КодВидаОС	Числовой	Длинное целое	Да
ВидОС	Текстовый	50	Да

### Нормы амортизации

<i>Наименование поля</i>	<i>Тип данных</i>	<i>Размер поля</i>	<i>Обязательное поле</i>
КодНормыАморт	Текстовый	8	Да
КодВидаОС	Числовой*	Длинное целое	
ПодвидОС	Текстовый	60	
НормаАмортизации	Числовой	Одинарное с плав. точкой	

- С помощью мастера подстановок обеспечьте для поля «КодВидаОС» возможность подстановки в него значений из аналогичного поля таблицы «Виды ОС». Список подстановок должен содержать также поле «ВидОС».

### Подразделения предприятия

<i>Наименование поля</i>	<i>Тип данных</i>	<i>Размер поля</i>	<i>Обязательное поле</i>
КодПодразделения	Числовой	Длинное целое	Да
Подразделение	Текстовый	60	Да
КодВышестоящПодр	Числовой	Длинное целое	
ВидПодразделения	Текстовый*	17	

- С помощью мастера подстановок обеспечьте для поля «ВидПодразделения» возможность подстановки в него следующих значений: Административное, Инженерное, Исследовательское, Питания, Производственное, Складское, Строительное, Торговое, Прочее.

- Обеспечьте для поля «КодВышестоящПодр» возможность подстановки в него значений из поля «КодПодразделения» этой же таблицы. Для этого сначала сохраните таблицу, а затем в окне конструктора таблицы раскройте вкладку «Подстановка» и в строке «Тип элемента управления» выберите значение «Поле со списком». Затем выполните соответствующую настройку значений строк, расположенных ниже.

### Материально ответственные

<i>Наименование поля</i>	<i>Тип данных</i>	<i>Размер поля</i>	<i>Обязательное поле</i>
КодМатОтв	Счетчик		
КодПодразделения	Числовой*	Длинное целое	
Фамилия	Текстовый	20	Да
Имя	Текстовый	20	
Отчество	Текстовый	20	

- С помощью мастера подстановок обеспечьте для поля «КодПодразделения» возможность подстановки в него значений из аналогичного поля таблицы «Подразделения предприятия». Список подстановок должен содержать также поле «Подразделение».

### Инвентарные карточки ОС

<i>Наименование поля</i>	<i>Тип данных</i>	<i>Размер поля</i>	<i>Обязательное поле</i>
ИнвентарныйНомер	Текстовый	10	Да
НаименованиеОС	Текстовый	80	Да
КодГруппыОС	Числовой*	Длинное целое	
КодНормыАморт	Текстовый*	8	
РежимИспользования	Текстовый*	29	
КодМатОтв	Числовой*	Длинное целое	
ДатаПоступления	Дата/время		Да
ДатаВыбытия	Дата/время		
ПервоначСтоимость	Денежный		Да
СуммаИзноса	Денежный		
УчитыватьПробег	Логический		
СодержаниеЗолота	Числовой	Одинарное с плав. точкой	
СодержаниеСеребра	Числовой	Одинарное с плав. точкой	
СодержаниеПлатины	Числовой	Одинарное с плав. точкой	
СодержПлатГруппы	Числовой	Одинарное с плав. точкой	

- С помощью мастера подстановок обеспечьте для поля «КодГруппыОС» возможность подстановки в него значений из аналогичного поля таблицы «Группы ОС». Список подстановок должен содержать также поле «ГруппаОС».

- С помощью мастера подстановок обеспечьте для поля «КодНормыАморт» возможность подстановки в него значений из аналогичного поля таблицы «Нормы амортизации». Список подстановок должен содержать также поля «КодВидаОС» и «ПодвидОС».

- С помощью мастера подстановок обеспечьте для поля «РежимИспользования» возможность подстановки в него следующих значений: В эксплуатации, В запасе или резерве, Сдано в аренду, Принято в аренду, Принято в долгосрочную аренду.

- С помощью мастера подстановок обеспечьте для поля «КодМатОтв» возможность подстановки в него значений из аналогичного поля таблицы «Материально ответственные». Список подстановок должен содержать также поля «КодПодразделения» и «Фамилия».

- Для полей «ДатаПоступления» и «ДатаВыбытия» выберите с помощью мастера маску ввода с названием «Краткий формат даты».

- Установите для поля «УчитыватьПробег» значение по умолчанию, равное False.

- Для полей «СодержаниеЗолота», «СодержаниеСеребра», «СодержаниеПлатины», «СодержПлатГруппы» установите число десятичных знаков равным 4.

## Пробег автотранспортных ОС

<u>Наименование поля</u>	<u>Тип данных</u>	<u>Размер поля</u>	<u>Обязательное поле</u>
КодАвтоПробега	Счетчик		
ИнвентарныйНомер	Текстовый*	10	Да
Месяц	Числовой	Байт	
Год	Числовой	Целое	
Пробег_км	Числовой	Одинарное с плав. точкой	

- С помощью мастера подстановок обеспечьте для поля «ИнвентарныйНомер» возможность подстановки в него значений из аналогичного поля таблицы «Инвентарные карточки ОС». Список подстановок должен содержать также поле «НаименованиеОС».

- Для поля «Месяц» установите условие на значение равным: Between 1 And 12

- Для поля «Месяц» установите сообщение об ошибке вводом следующего текста: «Введено неверное значение номера месяца».

- Для поля «Год» установите условие на значение равным: >= 2005 And < 2050

- Для поля «Год» установите сообщение об ошибке вводом следующего текста: «Введено неверное значение года».

- Для поля «Пробег\_км » установите формат поля равным: # ##0,0

В созданных выше таблицах найдите числовые поля и очистите их свойство «Значение по умолчанию» от нулей, которые Access подставляет автоматически при создании полей этого типа.

В каждой из созданных таблиц просмотрите еще раз названия полей и, если желаете, можете для некоторых из них указать более удобные названия, что позволяет использовать их в формах и отчетах вместо основных. Эти более удобные названия (более короткие или включающие в себя пробелы и т.д.) указываются в свойстве «Подпись». Например, «Инв. №» для «ИнвентарныйНомер» или «Код группы» для «КодГруппыОС» и т.д.

Откройте схему данных с помощью соответствующей кнопки на панели инструментов и настройте все созданные с помощью мастера подстановок связи между таблицами таким образом, чтобы была обеспечена целостность данных. Для этого нужно выделить связь и с помощью ее контекстного меню, вызываемого правой кнопкой мыши, открыть окно «Изменение связей», где и выполнить нужную настройку.

*Примечание.* Если в схеме данных какая-либо связь отсутствует, то ее необходимо создать путем перетаскивания первичного ключа из родительской таблицы на аналогичное поле дочерней таблицы, которое после этого становится ее внешним ключом.

В итоге схема данных должна выглядеть примерно так, как показано на рис. 1.1.

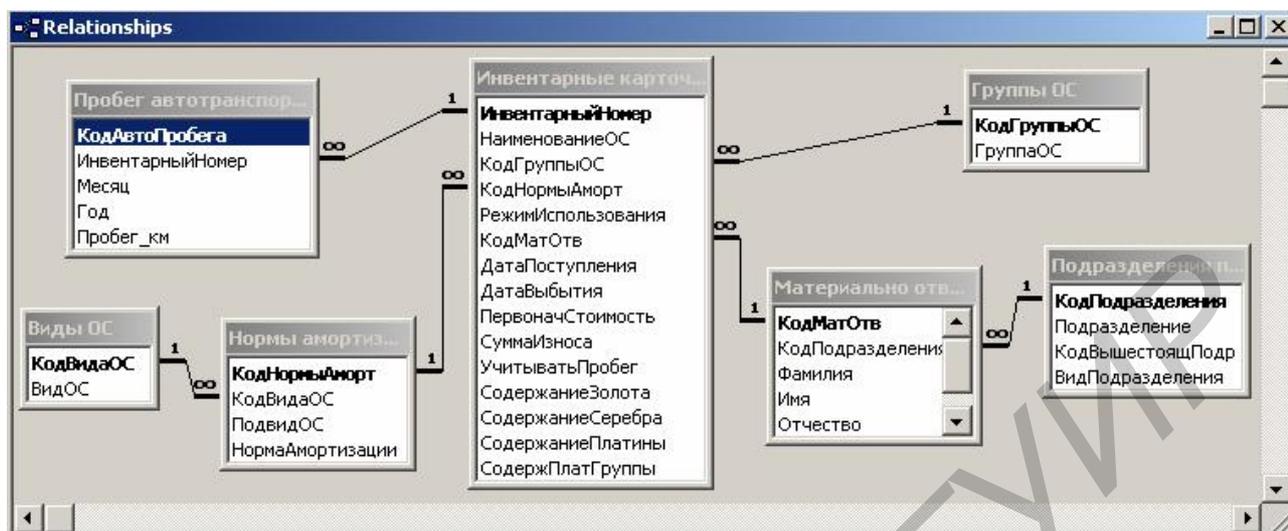


Рис. 1.1. Схема данных

Заполните таблицу «Группы ОС» следующими значениями:

ГруппаОС
Промышленно-производственные
Производственные других отраслей
Непроизводственные
Жилищного хозяйства

Заполните таблицу «Виды ОС» следующими значениями:

КодВидОС	ВидОС
1	Здания
2	Сооружения
3	Передаточные устройства
4	Силовые машины и оборудование
5	Рабочие машины и оборудование
6	Измерительные приборы и лабораторное оборудование
7	Вычислительная техника
8	Прочие машины и оборудование
9	Транспортные средства
10	Инструмент
11	Производств. и хоз. инвентарь и принадлежности
12	Прочие основные фонды

Заполните таблицу «Нормы амортизации» приведенными ниже значениями. При этом значение поля «КодВидОС» вместо ручного ввода выбирайте из выпадающего списка.

<i>КодНормыАморт</i>	<i>КодВидаОС</i>	<i>ПодвидОС</i>	<i>НормаАмортизации</i>
10001	1	Здания многоэтажные (более 2 этажей)	1
10002	1	Здания 2-этажные, кроме дерев., ж/б 1-этажные	1,2
10101	1	Жилые здания с кирпич. стенами, перекрытия ж/б	0,8
20200	2	Мосты ж/б, бетонные и каменные	1
20201	2	Мосты металлические	2
20226	2	Производственные автодороги, булыжные мостовые	8,3
41000	4	Металлорежущее оборудование с ручным управлением	5
44174	4	Автоматические линии и многоэлектрод. свар. аппараты	11,1
44804	7	Средства копирования и размножения документов	12,5
45613	8	Телеграфные и факсимильные аппараты	7,4
47024	6	Измерительные приборы, мед. диагностич. аппаратура	10,4
48000	7	Машины и комплексы электронные цифровые вычислит.	12,5
48001	7	Аналоговые клавишные ЭВМ и ПК	10
50402	9	Автомобили (более 2 т) с ресурсом до 200 тыс. км	0,37
50403	9	Автомобили (более 2 т) с ресурсом от 200 до 250 тыс. км	0,3
50404	9	Автомобили (более 2 т) с ресурсом от 250 до 350 тыс. км	0,2
50405	9	Автомобили (более 2 т) с ресурсом от 350 до 400 тыс. км	0,17
51400	9	Автомобили грузоподъемностью до 0,5 т	20
51401	9	Автомобили грузоподъемностью от 0,5 до 2 т	14,3
51415	9	Легковые автомобили (до 1,2 л)	18,2
51416	9	Легковые автомобили (от 1,2 до 1,8 л)	14,3
51418	9	Легковые автомобили (от 1,8 до 3,5 л)	11,1
60002	10	Прочий инструмент	20
70005	11	Электроарматура и электроприборы	8,3
70010	11	Стеллажи стоечные	5,6

Заполните таблицу «**Подразделения предприятия**» приведенными ниже значениями. При этом значение поля «ВидПодразделения» выбирайте из выпадающего списка.

<i>КодПодразделения</i>	<i>Подразделение</i>	<i>КодВышестоящПодр</i>	<i>ВидПодразделения</i>
10	Администрация		Административное
21	Склад	10	Складское
31	ОКБ	10	Инженерное
43	Сборочный цех	31	Производственное

Заполните таблицу «**Материально ответственные**» приведенными ниже значениями. При этом значение поля «КодПодразделения» выбирайте из выпадающего списка.

<i>КодПодразделения</i>	<i>Фамилия</i>	<i>Имя</i>	<i>Отчество</i>
10	Савельев	Федор	Николаевич
21	Прошкина	Татьяна	Ивановна
31	Юрьева	Вероника	Павловна
43	Тихонов	Владимир	Игоревич
43	Боброва	Ольга	Викторовна

Заполните таблицу «**Инвентарные карточки ОС**» данными, введя в нее не менее 15 – 20 записей. Наименования ОС могут быть, например, такие: «Автомобиль ГАЗ-53», «Здание гаража», «Мост ж/б арочный», «Копировальный автомат МТА-20», «Принтер лазерный Lexmark», «Компьютер NTT», «Фрезерный станок с ЧПУ», «Сверлильный станок», «Станок для правки и резки арматуры» и т. д. При этом значения полей «КодГруппыОС», «КодНормыАморт», «РежимИспользования», «КодМатОтв» выбирайте из соответствующих выпадающих списков.

Заполните таблицу «**Пробег автотранспортных ОС**» данными, введя в нее по несколько записей для каждого автотранспортного ОС. При этом значение поля «ИнвентарныйНомер» выбирайте из выпадающего списка.

**Внимание!** Каждый раз по окончании работы с СУБД Access для уменьшения размера файла базы данных выполняйте сжатие базы данных с помощью пункта меню **Сервис® Служебные программы® Сжать и восстановить базу данных**.

Библиотека БГУИР

## Лабораторная работа №2. Создание запросов

### Цель работы

1. Ознакомиться с процессом создания запросов в режиме конструктора.
2. Освоить способ создания формул и условий отбора данных с помощью построителя выражений.

### Задачи

1. Создание запроса на выборку с заданными типом запроса, перечнем полей в запросе, сортировкой полей, условием отбора данных. Или же создание запроса другого типа (с группировкой данных, перекрестного, параметрического, с вычисляемыми полями, на обновление, на создание таблицы, на удаление данных, на поиск записей без подчиненных).
2. Выполнение запроса с целью просмотра и анализа выбираемых запросом данных.
3. Просмотр SQL-кода запроса.

### Методические указания

Чаще всего запросы предназначены для выполнения всевозможных выборок информации из базы данных с использованием различных критериев отбора данных, хотя это и не единственное их назначение. Запросы можно создавать с помощью мастеров, с помощью конструктора, а также путем ручного ввода кода SQL.

Режим конструктора является наиболее универсальным и позволяет создавать запросы любого типа. В этом случае пользователь сам определяет тип создаваемого запроса и разрабатывает его структуру, а именно: поля из каких таблиц будут использоваться в запросе, каким условиям должны удовлетворять отбираемые данные, по каким полям будет выполняться сортировка и др.

В верхнюю часть окна конструктора добавляются все необходимые для формирования запроса таблицы.

Нижняя часть окна конструктора (бланк запроса) разбита на столбцы – по одному столбцу на каждое поле, включаемое в запрос. В каждом столбце представлены: имя поля и имя таблицы, которой оно принадлежит; флажок, указывающий на необходимость выполнения сортировки по данному полю; флажок, указывающий на необходимость вывода данного поля на экран, а также условие отбора данных, связанное с этим полем. Условия отбора данных представляют собой ограничения, налагаемые на значения полей, входящих в состав запроса. Сложные критерии отбора можно формировать с помощью построителя выражений.

Таким образом, добавив в верхнюю часть окна конструктора необходимые для построения запроса таблицы, а в бланк запроса – нужные поля из этих таблиц, а также задав необходимые условия отбора данных и способ сортировки,

мы создадим наиболее часто используемый запрос, называемый «Запрос на выборку», который можно выполнить и просмотреть, если нажать на панели инструментов кнопку **Запуск** (с пиктограммой в виде восклицательного знака). Разновидностями запроса на выборку являются параметрический запрос и запрос с вычисляемыми полями.

*Параметрический запрос* служит для расширения области применения запроса на выборку, так как в этом случае условие отбора уже не фиксируется в бланке запроса, а будет запрашиваться у пользователя после нажатия им кнопки **Запуск**. Для преобразования запроса на выборку в параметрический запрос необходимо заменить конкретное условие отбора данных соответствующей текстовой подсказкой, заключенной в квадратные скобки.

*Запрос с вычисляемыми полями* позволяет включать в запрос поля, реально не существующие в таблицах или запросах. Для преобразования запроса на выборку в запрос с вычисляемым полем необходимо в бланке запроса в верхней строке первого пустого столбца записать имя вычисляемого поля с двоеточием в конце. Далее ввести выражение – формулу, по которой будут выполняться вычисления. Для сложных формул можно воспользоваться построителем выражений. При этом имена полей нужно выбирать из соответствующих списков, а не набирать вручную. Также имеется широкий выбор встроенных функций (более 200).

Кроме запроса на выборку имеются еще следующие типы запросов:

1. *Запрос с группировкой данных*. Позволяет сгруппировать данные на основе значений одного или нескольких полей и для каждой группы подсчитать итоговые данные (сумму, среднее, минимум, максимум и др.). Для создания запроса этого типа нужно после открытия окна конструктора нажать на панели инструментов кнопку **Групповые операции** ( $\Sigma$ ). При этом в бланк запроса добавится строка «Групповая операция», в которой для каждого включенного в запрос поля нужно либо оставить значение «Группировка», либо выбрать (из выпадающего списка) нужную функцию для подсчета итоговых данных по этому полю.

2. *Перекрестный запрос* – это особый тип запроса с группировкой данных. При этом одно или более полей определяют своими значениями перечень строк запроса. Одно поле определяет своими значениями перечень столбцов запроса. Еще одно поле используется для подсчета итоговых данных, которые будут располагаться на местах пересечения строк и столбцов. После выбора этого типа запросов (с помощью кнопки **Тип запроса** на панели инструментов) в бланк запроса добавляются строки «Групповая операция» и «Перекрестная таблица». Первая из них настраивается так же, как и в запросе с группировкой данных, а вторая – задает назначение рассмотренных выше полей в перекрестном запросе (возможные значения: «Заголовки строк», «Заголовки столбцов», «Значение»).

3. *Запрос на создание таблицы*. Предназначен для создания новой таблицы на основе данных, выбираемых с помощью запроса. После выбора этого типа запросов пользователю предлагается ввести имя таблицы, которая будет создана. Далее формирование запроса ничем не отличается от формирования за-

проса на выборку. После нажатия кнопки **Запуск** появится сообщение о том, сколько записей будет содержать новая таблица.

4. *Запрос на добавление.* Предназначен для добавления данных из одной или нескольких таблиц в другую таблицу. У пользователя запрашивается имя таблицы, в которую будут добавлены данные. Кроме того, в бланке запроса вместо строки «Вывод на экран» появляется строка «Добавление», которая служит для того, чтобы сопоставить столбцы бланка запроса с полями таблицы-получателя. Далее формирование запроса ничем не отличается от формирования запроса на выборку. После нажатия кнопки **Запуск** появится сообщение о том, сколько записей будет добавлено в таблицу-получатель.

5. *Запрос на обновление.* Предназначен для внесения изменений в табличные данные. После выбора этого типа запросов в бланке запроса вместо строк «Сортировка» и «Вывод на экран» появляется строка «Обновление» для ввода в нее выражений, описывающих требуемое изменение данных. В бланк запроса необходимо включить только поля, значения которых будут обновляться, а также поля, используемые для задания условий отбора или используемые в выражениях, описывающих требуемое изменение данных. После нажатия кнопки **Запуск** появится сообщение о том, сколько записей будет обновлено.

6. *Запрос на удаление.* Предназначен для удаления данных из таблиц. После выбора этого типа запросов в бланке запроса вместо строк «Сортировка» и «Вывод на экран» появляется строка «Удаление». В бланк запроса необходимо включить только поля, используемые для задания условий отбора. Если нужно удалить все записи из таблицы, то бланк запроса будет пустым (сама таблица будет содержаться в верхней части окна конструктора). После нажатия кнопки **Запуск** появится сообщение о том, сколько записей будет удалено.

7. *Запрос на поиск повторяющихся записей* обычно создается с помощью специализированного мастера и позволяет выявить дублирование данных в таблице или запросе. При этом из списка полей выделяется подмножество полей, наборы значений которых и будут подвергаться проверке на совпадение. Если в запрос необходимо добавить условие отбора данных, то нужно воспользоваться режимом конструктора.

8. *Запрос на поиск записей, не имеющих подчиненных*, обычно создается с помощью специализированного мастера. При этом указываются две таблицы со связью «один ко многим». Запрос позволяет выявить те записи в родительской таблице, для которых отсутствуют подчиненные им записи в дочерней таблице.

Как уже говорилось, условие отбора данных представляет собой ограничение, налагаемое на значения поля, входящего в состав запроса. С точки зрения математической логики это предикат, то есть выражение логического типа, которое может принимать одно из двух значений: True (Истина) или False (Ложь). Те записи, для которых значение предиката есть истина, будут включены в результирующее множество строк.

Например:

```
Фамилия = "Иванов"  
СуммаИзноса > 100000  
ДатаВыбытия = "05.18.2007"
```

Предикаты обычно включают в себя имена полей, константы, операторы сравнения (=, <>, <, <=, >, >=) и логические связки NOT, AND, OR (соответственно НЕ, И, ИЛИ), которые перечислены в порядке убывания их старшинства. В свою очередь операторы сравнения имеют более высокий приоритет, чем логические связки (что часто позволяет обходиться без использования круглых скобок).

Например:

```
Фамилия = "Иванов" OR Фамилия = "Петров"  
СуммаИзноса >= 100000 AND СуммаИзноса <= 200000
```

Необходимо отметить, что имя поля можно в предикате не указывать, если только этот предикат принадлежит столбцу, связанному с этим же самым полем. Поэтому если предыдущие два предиката связаны с полями «Фамилия» и «СуммаИзноса» соответственно, то их можно записать компактнее:

```
"Иванов" OR "Петров"  
>= 100000 AND <= 200000
```

Предикаты могут включать в себя операторы IN и BETWEEN. Первый из них задает множество значений, а второй – интервал значений, которые и определяют критерии отбора данных. Оператор BETWEEN всегда используется со связкой AND, разделяющей граничные значения интервала. Поэтому рассмотренные выше предикаты можно переписать так:

```
Фамилия IN ("Иванов"; "Петров")  
СуммаИзноса BETWEEN 100000 AND 200000
```

или в сокращенной форме:

```
IN ("Иванов"; "Петров")  
BETWEEN 100000 AND 200000
```

Для текстовых полей в предикатах может использоваться оператор LIKE, который задает строку символов в качестве критерия отбора данных. Эта строка может включать в себя метасимволы «\*» и «?». Первый из них обозначает любое число любых символов, а второй – любой один символ. Например:

```
Фамилия LIKE "А*"  
Фамилия LIKE "Я??о*ий"  
Фамилия LIKE "И*ов" OR Фамилия LIKE "П*ов"
```

или в сокращенной форме:

```
"А*"  
"Я??о*ий"  
"И*ов" OR "П*ов"
```

(обнаружив метасимволы, Access сам подставит оператор LIKE).

Зарезервированное слово NULL, которое означает, что значение поля не известно, используется в предикатах всегда со связкой IS. Например:

```
ДатаВыбытия IS NULL
```

СуммаИзноса IS NOT NULL

или в сокращенной форме:

NULL

NOT NULL

(Access сам подставит связку IS).

Необходимо отметить, что все предикаты, которые размещены в бланке запроса в одной и той же строке, Access будет соединять между собой связками AND. Если же предикаты размещены и в строках, расположенных ниже, то сначала будут выполнены все построчные соединения связками AND, и лишь затем полученные выражения будут соединены между собой связками OR. В итоге будет сформировано результирующее условие отбора данных.

Обычно запросы создаются в режиме визуального проектирования, однако можно создать любой запрос и путем ввода соответствующего SQL-кода. Для этого необходимо:

- 1) открыть окно конструктора, не добавляя в него ни одной таблицы;
- 2) выбрать в выпадающем списке кнопки **Вид** строку «Режим SQL»;
- 3) в появившемся окне ввести текст SQL-команды;
- 4) сохранить запрос.

В принципе не важно, создается ли запрос в режиме визуального проектирования или путем ручного ввода кода SQL, однако последний способ позволяет создавать и такие запросы, которые нельзя создать в визуальном конструкторе (например, запросы на объединение данных, запросы управления транзакциями, запросы к серверу и др.).

### **Варианты заданий**

1. Создайте запрос с именем «**Запрос на выборку 1**» со следующими свойствами.

*Тип запроса:* запрос на выборку.

*Перечень полей в запросе:* ГруппаОС, ИнвентарныйНомер, НаименованиеОС, Фамилия, Имя, Отчество, НормаАмортизации, УчитыватьПробег, ПервоначСтоимость, СуммаИзноса.

*Сортировка:* ГруппаОС (по убыванию), НаименованиеОС (по возрастанию).

*Условие отбора данных:* все записи, для которых значение поля «НормаАмортизации» менее 0,5, или значение поля «ПервоначСтоимость» лежит в диапазоне от 20 000 000 до 100 000 000 и значение поля «СуммаИзноса» превышает 200 000.

2. Создайте запрос с именем «**Запрос на выборку 2**» со следующими свойствами.

*Тип запроса:* запрос на выборку.

*Перечень полей в запросе:* ИнвентарныйНомер, НаименованиеОС, КодПодразделения, Подразделение, КодМатОтв, Фамилия, ПервоначСтоимость, СодержаниеЗолота, СодержаниеСеребра, СодержаниеПлатины, СодержПлатГруппы.

*Сортировка:* СодержаниеЗолота (по убыванию), СодержаниеСеребра (по убыванию).

*Условие отбора данных:* все записи, для которых поле «СодержаниеЗолота» содержит ненулевое значение, или значение поля «СодержаниеСеребра» больше 3 и при этом хотя бы в одном из полей «СодержаниеПлатины», «СодержаниеПлатГруппы» содержится ненулевое значение.

3. Создайте запрос с именем **«Запрос на выборку 3»** со следующими свойствами.

*Тип запроса:* запрос на выборку.

*Перечень полей в запросе:* КодВидаОС, ВидОС, ПодвидОС, ИнвентарныйНомер, НаименованиеОС, КодПодразделения, Подразделение, ПервоначСтоимость, КодНормыАморт, НормаАмортизации, СуммаИзноса.

*Сортировка:* КодНормыАморт (по возрастанию), ПервоначСтоимость (по убыванию).

*Условие отбора данных:* все записи, для которых значение поля «НаименованиеОС» включает в себя сочетание букв «ан» начиная с третьей или последующих позиций букв, или в поле «ДатаВыбытия» содержится конкретная дата.

4. Создайте запрос с именем **«Запрос параметрический 1»** со следующими свойствами.

*Тип запроса:* запрос на выборку.

*Перечень полей в запросе:* ПодвидОС, ИнвентарныйНомер, НаименованиеОС, КодПодразделения, Подразделение, ПервоначСтоимость, НормаАмортизации, СуммаИзноса.

*Сортировка:* Подразделение (по возрастанию), ПервоначСтоимость (по убыванию).

*Условие отбора данных:* все записи, для которых значение поля «НормаАмортизации» меньше значения, введенного пользователем с клавиатуры, и значение поля «ДатаВыбытия» не определено.

5. Создайте запрос с именем **«Запрос параметрический 2»** со следующими свойствами.

*Тип запроса:* запрос на выборку.

*Перечень полей в запросе:* КодПодразделения, Подразделение, КодМатОтв, Фамилия, Имя, Отчество, ИнвентарныйНомер, НаименованиеОС, РежимИспользования, ПервоначСтоимость, СуммаИзноса.

*Сортировка:* Фамилия (по возрастанию), НаименованиеОС (по возрастанию).

*Условие отбора данных:* все записи, для которых значение поля «ДатаПоступления» попадает в указанный пользователем интервал дат.

6. Создайте запрос с именем **«Запрос с текстовым вычисляемым полем»** со следующими свойствами.

*Тип запроса:* запрос на выборку.

*Перечень полей в запросе:* ИнвентарныйНомер, НаименованиеОС, КодПодразделения, Подразделение, КодМатОтв, Фамилия, Имя, Отчество, ФИО.

Здесь поле «ФИО» является вычисляемым. Оно составляется из полей «Фамилия», «Имя» и «Отчество», разделяемых пробелами (для соединения в одном выражении текстовых полей и констант используйте оператор «&»).

*Сортировка:* ФИО (по возрастанию).

*Усложненный вариант:* вычисляемое поле составляется из фамилии и инициалов (начальных букв имени и отчества, дополненных точками).

7. Создайте запрос с именем «**Запрос с числовым вычисляемым полем**» со следующими свойствами.

*Тип запроса:* запрос на выборку.

*Перечень полей в запросе:* КодВидаОС, ВидОС, ИнвентарныйНомер, НаименованиеОС, КодПодразделения, КодНормыАморт, НормаАмортизации, ПервоначСтоимость, СуммаИзноса, ОстаточнаяСтоимость, СрокАмортизации. Здесь поля «ОстаточнаяСтоимость» и «СрокАмортизации» являются вычисляемыми. Первое из них должно отображать разность между первоначальной стоимостью и суммой износа, а второе основывается на формуле

*СрокАмортизации:*  $\text{Int}((\text{ПервоначСтоимость} - \text{СуммаИзноса}) / (\text{ПервоначСтоимость} * \text{НормаАмортизации} / 100 / 12))$

*Сортировка:* НормаАмортизации (по возрастанию), ОстаточнаяСтоимость (по убыванию).

8. Создайте запрос с именем «**Запрос с группировкой**» со следующими свойствами.

*Тип запроса:* групповая операция.

*Перечень полей в запросе:* ГруппаОС, ВидОС, ИнвентарныйНомер, ПервоначСтоимость, СуммаИзноса.

*Групповая операция:* для полей «ГруппаОС» и «ВидОС» – Группировка; для поля «ИнвентарныйНомер» – Count; для полей «ПервоначСтоимость», «СуммаИзноса» – Sum.

*Условие отбора данных:* инвентарные карточки всех тех основных средств, которые содержат хотя бы один из драгметаллов (это условие отбора требует отдельного столбца в бланке запроса и создается по аналогии с созданием вычисляемого поля). Групповая операция – Условие (Where – для англоязычной версии).

*Сортировка:* СуммаИзноса (по убыванию).

9. Создайте запрос с именем «**Запрос перекрестный**» со следующими свойствами.

*Тип запроса:* перекрестный запрос.

*Перечень полей в запросе:* ГруппаОС, ВидОС, РежимИспользования, СуммаИзноса.

*Групповая операция:* для полей «ГруппаОС», «ВидОС», «РежимИспользования» – Группировка; для поля «СуммаИзноса» – Sum.

*Перекрестная таблица:* для полей «ГруппаОС» и «ВидОС» – Заголовки строк; для поля «РежимИспользования» – Заголовки столбцов; для поля «СуммаИзноса» – Значение.

*Сортировка:* ГруппаОС (по убыванию), ВидОС (по возрастанию).

*Условие отбора данных:* инвентарные карточки всех тех основных средств, которые поступили на предприятие за последние 5 лет (это условие отбора требует отдельного столбца в бланке запроса и создается по аналогии с созданием вычисляемого поля). Групповая операция – Условие (Where – для англоязычной версии).

10. Создайте запрос с именем **«Запрос на обновление»** со следующими свойствами.

*Тип запроса:* запрос на обновление.

*Обновление:* изменить значение поля «СуммаИзноса» в таблице «Инвентарные карточки ОС» на значение, вычисляемое по следующей формуле:

$$\text{СуммаИзноса} + \text{ПервоначСтоимость} * \text{НормаАмортизации} / 100 / 12$$

*Условие отбора данных:* все записи, для которых значение поля «УчитыватьПробег» принимает значение False и значение поля «ОстаточнаяСтоимость» больше нуля.

Здесь поле «ОстаточнаяСтоимость» является вычисляемым и должно отображать разность между первоначальной стоимостью и суммой износа.

11. Создайте запрос с именем **«Запрос на обновление параметрический»** со следующими свойствами.

*Тип запроса:* запрос на обновление.

*Обновление:* изменить значение поля «СуммаИзноса» в таблице «Инвентарные карточки ОС» на значение, вычисляемое по следующей формуле:

$$\text{СуммаИзноса} + \text{ПервоначСтоимость} * \text{НормаАмортизации} / 100 * [\text{Укажите месячный пробег в км}] / 1000$$

*Условие отбора данных:* все записи, для которых значение поля «УчитыватьПробег» принимает значение True и значение поля «ОстаточнаяСтоимость» больше нуля.

Здесь поле «ОстаточнаяСтоимость» является вычисляемым и должно отображать разность между первоначальной стоимостью и суммой износа.

12. Создайте запрос с именем **«Запрос на создание таблицы»** со следующими свойствами.

*Тип запроса:* запрос на создание таблицы.

*Перечень полей в запросе:* ВидОС, ИнвентарныйНомер, НаименованиеОС, Подразделение, ВидПодразделения, СодержаниеЗолота, СодержаниеСеребра, СодержаниеПлатины, СодержПлатГруппы.

*Сортировка:* СодержаниеПлатины (по убыванию), СодержПлатГруппы (по убыванию), СодержаниеЗолота (по убыванию), СодержаниеСеребра (по убыванию).

*Условие отбора данных:* инвентарные карточки всех тех основных средств, которые содержат хотя бы один из драгметаллов.

13. Создайте запрос с именем **«Запрос на удаление»** со следующими свойствами.

*Тип запроса:* запрос на удаление.

*Условие отбора данных:* все записи из таблицы «Инвентарные карточки ОС», для которых значение поля «КодНормыАморт» начинается комбинацией цифр

101 или 20 и значение поля «СуммаИзноса» больше значения, введенного пользователем с клавиатуры.

14. Создайте запрос с именем **«Запрос на поиск записей без подчиненных»** со следующими свойствами.

*Тип запроса:* записи без подчиненных.

*Родительская таблица:* Нормы амортизации.

*Дочерняя таблица:* Инвентарные карточки.

*Перечень полей в запросе:* все поля, входящие в родительскую таблицу.

*Сортировка:* НормаАмортизации (по убыванию).

*Условие отбора данных:* все записи, для которых значение поля «КодВидаОС» равно 9 и значение поля «НормаАмортизации» попадает в интервал от 10 до 20.

15. Создайте запрос с именем **«Запрос с усложненной выборкой»** со следующими свойствами.

*Тип запроса:* запрос на выборку.

*Перечень полей в запросе:* ВидОС, ИнвентарныйНомер, НаименованиеОС, КодПодразделения, Подразделение, ПервоначСтоимость, СуммаИзноса, Год, Месяц, Пробег\_км.

*Сортировка:* Год (по убыванию), Месяц (по убыванию), НаименованиеОС (по возрастанию).

*Условие отбора данных:* все записи, для которых значение поля «Пробег\_км» лежит в диапазоне от 1 000 до 10 000 и которые удовлетворяют условию:

$\text{ПервоначСтоимость} - \text{СуммаИзноса} > \text{ПервоначСтоимость}$

$* \text{НормаАмортизации} / 2$

Это условие отбора требует отдельного столбца в бланке запроса и создается по аналогии с созданием вычисляемого поля.

## Лабораторная работа №3. Создание форм

### Цель работы

1. Ознакомиться с процессом создания форм в режиме конструктора.
2. Освоить использование панели элементов для добавления в форму элементов управления.
3. Изучить работу с мастерами панели элементов с целью создания простейших обработчиков событий.
4. Ознакомиться со способами конструирования составных форм.

### Задачи

1. Создание формы и связывание ее с таблицей или запросом.
2. Размещение на форме элементов управления.
3. Настройка вида формы, ее свойств и свойств элементов управления с помощью окна свойств.
4. Вставка в форму фрагментов графики и/или элементов управления ActiveX.
5. При необходимости (зависит от варианта задания) создание с помощью мастера панели элементов простейшего обработчика событий или составной формы.

### Методические указания

Формы служат для ввода и изменения данных в таблицах, а также для отображения в удобном виде результатов выполнения запросов. Обычно форма создается так, чтобы выглядеть как соответствующий ей первичный или сводный документ, и часто дополняется рисунками, диаграммами, поясняющими надписями, поэтому работать с формой намного удобнее и надежнее, чем непосредственно с таблицей.

Формы можно создавать в режиме автоформы, с помощью мастера и в режиме конструктора. Наиболее универсальным, но и более сложным является режим конструктора, в котором пользователь сам размещает в создаваемой форме все необходимые элементы управления и настраивает их свойства, а также настраивает свойства формы с помощью окна свойств.

Все элементы управления, размещаемые в форме, а также форма и ее разделы (заголовок, область данных, примечание) имеют объектно-ориентированную природу, в основе которой лежит понятие объекта. Каждый объект характеризуется своими свойствами и методами. Свойства объекта определяют различные характеристики элемента управления, а методы определяют те действия, которые можно выполнить в отношении объекта.

Кроме того, в результате действий пользователя, программы или операционной системы объекту могут быть посланы сообщения, которые называются событиями. События указывают объекту на необходимость выполнить тот или

иной метод, если только этот метод имеется. Расширение функциональных возможностей при работе с формой как раз и связано с разработкой дополнительных методов, управляемых событиями. Такие методы называются обработчиками событий.

Окно свойств, которое можно открыть в режиме конструктора форм, предназначено для просмотра и изменения свойств активного в данный момент объекта (элемента управления, формы или ее раздела).

В окне свойств имеется также вкладка «События», где перечисляются события, распознаваемые объектом. Если необходима реакция объекта на какое-либо событие, то следует создать метод, обрабатывающий это событие, либо с помощью конструктора макросов (последовательностей макрокоманд), либо написав соответствующую программу на языке VBA.

Таким образом, создание формы в режиме конструктора включает в себя следующие шаги.

1. С помощью окна свойств устанавливается связь формы с таблицей или запросом. Для этого настраивается свойство «Источник записей», расположенное на вкладке «Данные». После этого на экране появится окно «Список полей» с перечнем полей выбранного источника записей.

2. Если в форме должно отображаться более одной записи, то вид формы нужно поменять с обычного на ленточный или табличный с помощью свойства «Режим по умолчанию», расположенного на вкладке «Макет». При этом в форме будет отображаться столько записей, сколько позволят размеры области данных формы.

3. В окне «Список полей» выделяются все необходимые поля источника записей, которые путем перетаскивания мышью копируются в область данных формы, после чего размещаются там в нужных местах. Если на форме необходимо разместить еще какие-либо элементы управления, то нужно воспользоваться панелью элементов, при этом многие из элементов управления могут быть настроены на выполнение определенных функций с помощью соответствующих мастеров, которые запускаются сразу после размещения элементов управления на форме (запуск мастеров обеспечивается при нажатой кнопке **Мастера**, также расположенной на панели элементов).

4. При необходимости форма снабжается разделами «Заголовок формы» и «Примечание формы», куда также можно поместить нужные элементы управления и настроить их свойства.

5. При необходимости те или иные элементы управления, а также форма снабжаются обработчиками событий в виде макросов или программ на языке VBA. Для этого в окне свойств на вкладке «События» находится нужное событие и в правой части выбранной строки нажимается кнопка с тремя точками, после чего выбирается переход в конструктор макросов или в среду программирования VBA.

СУБД Access позволяет создавать составную форму, в которой одна, подчиненная форма находится внутри другой, главной формы. Если источниками записей для главной и подчиненной форм являются таблицы, связанные друг с

другом отношением «один ко многим» или «один к одному», то при перемещении по записям в главной форме в подчиненной форме выводятся только связанные записи.

Подчиненная форма вставляется в главную форму при помощи элемента управления «Подчиненная форма/отчет» панели элементов. При этом запускается мастер, который позволяет либо выбрать в качестве подчиненной уже существующую форму, либо задать список полей подчиненной таблицы, на основе которых будет создана подчиненная форма.

### **Варианты заданий**

1. Для запроса «Запрос на выборку 1» создайте в режиме конструктора форму **«фрм1: Просмотр запроса на выборку 1»** для последовательного просмотра на экране строк этого запроса, располагая поля в столбец. Задайте для формы какой-либо фон, используя окно свойств и, в частности, свойство «Рисунок» на вкладке «Формат». Добавьте также в форму вычисляемые поля «ФИО» и «Остаточная стоимость». Первое из них составляется из полей «Фамилия», «Имя» и «Отчество», разделяемых пробелами (для соединения в одном выражении текстовых полей и констант используйте оператор «&»). Второе поле должно отображать разность между первоначальной стоимостью и суммой износа. Используя окно свойств, оформите вычисляемые поля серым цветом.

2. Для запроса «Запрос на выборку 2» создайте в режиме конструктора форму **«фрм2: Просмотр запроса на выборку 2»** для последовательного просмотра на экране строк этого запроса, располагая поля в столбец. Оформите построенную форму каким-либо фоном с помощью кнопки **Автоформат**. Кроме того, уберите с формы стандартные кнопки навигации по строкам набора данных, после чего создайте аналогичные кнопки, используя панель элементов.

*Усложненный вариант.* Устраните появление на экране сообщений, сигнализирующих о попытке переместиться на несуществующую строку (т.е. на строку перед первой или после последней).

3. Для запроса «Запрос на выборку 3» создайте в режиме конструктора форму **«фрм3: Просмотр запроса на выборку 3»** для последовательного просмотра на экране строк этого запроса, располагая поля в столбец. Задайте для формы какой-либо фон, используя окно свойств и, в частности, свойство «Рисунок» на вкладке «Формат». Далее отключите системную кнопку закрытия формы, расположенную в ее верхнем правом углу, после чего разместите на форме кнопку с аналогичной функцией. Кнопку снабдите пиктограммой, а также подробной всплывающей подсказкой.

4. Создайте основную форму **«фрм4: Подвиды ОС»**, расположив на ней элемент управления типа список для просмотра всех подвидов основных средств и выбора одного из них. Создайте также вспомогательную форму

«**фрм4: Инвентарные карточки ОС**», предназначенную для последовательного ввода и редактирования данных в таблицу «Инвентарные карточки ОС». Разместите на основной форме кнопку, нажатие на которую должно выводить на экран вспомогательную форму, отображающую все инвентарные карточки ОС, код нормы амортизации которых совпадает с кодом нормы амортизации подвида основных средств, выбранным в списке основной формы.

5. Для таблицы «Подразделения предприятия» создайте форму «**фрм5: Подразделения предприятия**» для последовательного ввода и редактирования данных в эту таблицу. Далее разместите на форме поле для ввода даты и кнопку с надписью (или с всплывающей подсказкой) «Открыть запрос». После ввода какой-либо даты нажатие на кнопку должно открывать запрос, выбирающий все инвентарные карточки ОС, относящиеся к подразделению, указанному в соответствующем поле формы, и с датой поступления, не меньшей, чем дата, введенная в поле ввода даты.

6. Для запроса «Запрос с текстовым вычисляемым полем» создайте в режиме конструктора форму «**фрм6: Просмотр запроса с текстовым вычисляемым полем**» для последовательного просмотра на экране строк этого запроса. Оформите построенную форму каким-либо фоном с помощью кнопки **Автоформат**, а вычисляемые поля выделите каким-либо цветом. Далее разместите на форме кнопку, нажатие на которую должно открывать в отдельном окне упомянутый выше запрос, строки которого и отображает данная форма.

7. Для запроса «Запрос с числовым вычисляемым полем» создайте в режиме конструктора форму «**фрм7: Просмотр запроса с числовым вычисляемым полем**» для последовательного просмотра на экране строк этого запроса. Задайте для формы какой-либо фон, используя окно свойств и, в частности, свойство «Рисунок» на вкладке «Формат», а вычисляемые поля выделите каким-либо цветом. Далее разместите на форме кнопку, нажатие на которую должно открывать окно поиска записи. Снабдите кнопку подробной всплывающей подсказкой, а также текстом, отображаемым в строке состояния.

8. Для таблицы «Инвентарные карточки ОС» создайте форму «**фрм8: Инвентарные карточки ОС**» для последовательного ввода и редактирования данных в эту таблицу. В форму вставьте дополнительные элементы управления, в частности, календарь вместо поля «ДатаПоступления» (используйте кнопку **Другие элементы** на панели элементов). Вставьте также свободную рамку объекта (с помощью соответствующей кнопки на панели элементов) с помещением туда рисунка Paintbrush для последующего изображения на нем, например, логотипа фирмы. Календарь свяжите с полем «ДатаПоступления», используя окно свойств и его вкладку «Данные». Примерный вид формы в режиме просмотра изображен на рис. 3.1.

Фрм8: Инвентарные карточки ОС : Form

Инвентарный №: 3270

Наименование ОС: Рабочая станция VAX-2201

Код группы ОС: 3

Код вида ОС: 48000

Режим использования: Принято в долгосрочную аренду

Код мат. отв.: 3

Дата поступления: **дек 2005** дек 2005

Пн	Вт	Ср	Чт	Пт	Сб	Вс
28	29	30	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	
2	3	4	5	6	7	8

Первонач. стоим.: 95 287 600,00р.

Сумма износа: 11 934 539,00р.

Учитывать пробег

Содержание драйверов:

	5,629	0,0167	
монитор	сервер	принтер	латинская группа

Record: 7 of 13

Рис. 3.1. Форма «Фрм8: Инвентарные карточки ОС»

9. Создайте в режиме конструктора составную форму, предназначенную для ввода и редактирования данных в родительскую таблицу «Группы ОС» и дочернюю таблицу «Инвентарные карточки ОС». Для этого сначала создайте вспомогательную ленточную форму «Фрм9: Ленточные инвентарные карточки» для ввода данных в дочернюю таблицу (ленточный тип устанавливается в окне свойств формы: вкладка «Макет», строка «Режим по умолчанию»). Далее создайте основную форму «Фрм9: Группы – Инвентарные карточки» для ввода данных в родительскую таблицу. Затем в нижнюю часть этой формы вставьте вспомогательную ленточную форму с помощью кнопки **Подчиненная форма/отчет**, расположенной на панели элементов. В итоге составная форма в режиме просмотра должна выглядеть так, как изображено на рис. 3.2.

Фрм9: Группы – Инвентарные карточки : Form

Код группы ОС:

Группа ОС:

**фрм9: Ленточные инвентарные карточки**

Инв.№	Наименование	КодГр	КодНормыАм	Реж.использования	КодМатОт	ДатаПост	ДатаВыб	Перв.с
178	Автомобиль ГАЗ-53	1	50402	В эксплуатации	1	21.07.2003		22 24
179	Автомобиль VWЛТ28	1	50403	В эксплуатации	2	23.08.2004		38 46
2673	Здание гаража	1	10002	В эксплуатации	2	16.11.2000		26 40
920	Фрезерный станок	1	41000	В запасе или рез	4	29.12.2001		27 36
922	Станок для правки и резки арматур	1	41000	В эксплуатации	4	30.05.2003		13 45
924	Сверлильный станок	1	41000	В запасе или рез	4	06.06.2004		8 67
930	Фрезерный станок с ЧПУ	1	41000	В эксплуатации	5	04.08.2004		28 64

Record: 1 of 8

Record: 1 of 4

Рис. 3.2. Форма «фрм9: Группы – Инвентарные карточки»

10. Создайте в режиме конструктора составную форму, предназначенную для ввода и редактирования данных в родительскую таблицу «Виды ОС» и дочернюю таблицу «Нормы амортизации». Для этого сначала создайте вспомогательную табличную форму «фрм10: Табличные нормы амортизации» для ввода данных в дочернюю таблицу (табличный тип формы устанавливается с помощью окна свойств выбором вкладки «Макет» и строки «Режим по умолчанию»). Далее создайте основную форму «фрм10: Виды – Нормы амортизации» для ввода данных в родительскую таблицу. Затем в нижнюю часть этой формы вставьте вспомогательную табличную форму с помощью кнопки **Подчиненная форма/отчет**, расположенной на панели элементов. В итоге составная форма в режиме просмотра должна выглядеть так, как показано на рис. 3.3.

Фрм10: Виды – Нормы амортизации : Form

Код вида ОС:

Вид ОС:

**фрм10: Табличные нормы амортизации**

Код нормы	Код вида	Подвид ОС:	Норма амортиза
50402	9	Автомобили(более 2 т) с ресурсом до 200 тыс. Км	0,37
50403	9	Автомобили(более 2 т) с ресурсом от 200 до 250 тыс. Км	0,3
50404	9	Автомобили(более 2 т) с ресурсом от 250 до 350 тыс. Км	0,2
50405	9	Автомобили(более 2 т) с ресурсом от 350 до 400 тыс. Км	0,17
51400	9	Автомобили грузоподъемностью до 0.5 т	20
51401	9	Автомобили грузоподъемностью от 0.5 до 2 т	14,3
51415	9	Легковые автомобили (до 1.2 л)	18,2

Record: 1 of 9

Record: 9 of 12

Крепкая национальная валюта - залог успехов в экономике!

Рис. 3.3. Форма «фрм10: Виды – Нормы амортизации»

11. Создайте в режиме конструктора составную форму, предназначенную для ввода и редактирования данных в родительскую таблицу «Нормы Амортизации» и дочернюю таблицу «Инвентарные карточки ОС». Для этого сначала создайте вспомогательную форму «**фрм11: Инвентарные карточки ОС**» для ввода данных в дочернюю таблицу, располагая поля ввода в 3 столбца. Далее создайте основную форму «**фрм11: Нормы Амортизации – Инвентарные карточки**» для ввода данных в родительскую таблицу. Затем в нижнюю часть этой формы вставьте вспомогательную форму с помощью кнопки **Подчиненная форма/отчет**, расположенной на панели элементов. В итоге составная форма в режиме просмотра должна выглядеть так, как показано на рис. 3.4.

Рис. 3.4. Форма «фрм11: Нормы Амортизации – Инвентарные карточки»

12. Создайте в режиме конструктора составную форму, предназначенную для ввода и редактирования данных в родительскую таблицу «Подразделения предприятия», в дочернюю таблицу «Материально ответственные» и в субдочернюю таблицу «Инвентарные карточки ОС». Для этого сначала создайте вспомогательную табличную форму «**фрм12: Инвентарные карточки ОС**» для ввода данных в субдочернюю таблицу (табличный тип формы устанавливается с помощью окна свойств выбором вкладки «Макет» и строки «Режим по умолчанию»). Затем создайте вспомогательную составную форму «**фрм12: Мат\_ответственные – Инвентарные карточки**» для ввода данных в дочернюю таблицу. В нижнюю часть этой формы вставьте вспомогательную табличную форму с помощью кнопки **Подчиненная форма/отчет**, расположенной на панели элементов. Далее создайте основную форму «**фрм12: Подразделения – Мат\_отв – Инв\_карточки**» для ввода данных в родительскую таблицу. В нижнюю часть этой формы вставьте вспомогательную составную форму описанным выше способом. В итоге основная форма в режиме просмотра должна выглядеть так, как показано на рис. 3.5.

Фрм12: Подразделения - Мат\_отв - Инв\_карточки: Form

код подразделения: 10 | Администрация | код вышест. подр.: | Административное | вид подразделения:

Фрм12: Мат\_ответственные - Инвентарные карточки

Код мат.отв.: 1 | Фамилия: Савельев  
 Код подразделения: 10 | Имя: Федор  
 Отчество: Николаевич

Фрм12: Инвентарные карточки ОС

Инв. №	Наименование ОС:	Код груп	Код нормы	Режим испол.	Код мат.от	Дата поступ	Дата выбыт	Первонач
178	Автомобиль ГАЗ-53	1	50402	В эксплуа	1	21.07.2003		22 2
180	Автомобиль AUDI	1	51416	В эксплуа	1	17.06.2004		99 3
201	Мост ж/б арочный	2	20200	Принято в	1	02.08.2005		87 3
*					1			

Record: 1 of 3

Record: 1 of 2

Record: 1 of 4

Рис. 3.5. Форма «Фрм12: Подразделения – Мат\_отв – Инв\_карточки»

13. Для таблицы «Инвентарные карточки ОС» создайте основную форму «Фрм13: Инвентарные карточки ОС» для последовательного ввода и редактирования данных в эту таблицу. Поля формы расположите в 2 столбца. Далее создайте вспомогательную форму «Фрм13: Материально ответственные», которая предназначена для редактирования и добавления записей в таблицу «Материально ответственные». В основную форму вставьте кнопки **Открыть** и **Фильтровать**. Первая из них предназначена для открытия вспомогательной формы. Вторая кнопка также должна открывать вспомогательную форму, однако показывать в ней только ту запись, в которой значение поля «КодМатОтв» совпадает со значением аналогичного поля в основной форме.

14. Создайте в режиме мастера форму «Фрм14: Диаграмма износа ОС по мат\_ответственным» для вывода на экран диаграммы, отображающей для таблицы «Инвентарные карточки ОС» суммы износа основных средств по каждому материально ответственному в разрезе режимов использования основных средств. В итоге форма «Фрм14: Диаграмма износа ОС по мат\_ответственным» должна выглядеть примерно так, как показано на рис. 3.6.

15. В таблицу «Материально ответственные» добавьте поле «Фото» с типом данных «Поле объекта OLE». В режиме просмотра таблицы «Материально ответственные» заполните в каждой записи поле «Фото», используя команду меню **Вставка® Объект** и выбирая картинки из файлов: Фото1.bmp, Фото2.bmp, ..., Фото6.bmp. Далее для таблицы «Материально ответственные» создайте форму «Фрм15: Материально ответственные» для ввода и редактирования данных в эту таблицу, располагая поля ввода в столбец. Размер поля «Фото» установите примерно 2 x 1,5 см. Оформите построенную форму каким-либо фоном с помощью кнопки **Автоформат**.

*Примечание.* По окончании работы для уменьшения размера файла базы данных очистите поле «Фото» во всех записях таблицы «Материально ответственные» и далее выполните сжатие базы данных с помощью пункта меню Сервис® Служебные программы® Сжать и восстановить базу данных.



Рис. 3.6. Форма «фрм14: Диаграмма износа ОС по мат\_ответственным»

## Лабораторная работа №4. Создание отчетов

### Цель работы

1. Ознакомиться с процессом создания отчетов в режиме конструктора.
2. Освоить операцию группирования данных и способы выполнения вычислений в отчетах.
3. Ознакомиться со способами конструирования составных отчетов.

### Задачи

1. Создание отчета и связывание его с таблицей или запросом.
2. Группирование данных с подсчетом по числовым полям итогов для каждой группы и для всего отчета в целом.
3. При необходимости (зависит от варианта задания) создание составного отчета с помощью мастера панели элементов.

### Методические указания

Несмотря на то что форму, как и таблицу или запрос, можно распечатать на принтере с помощью команды меню **Файл® Печать**, в Access имеется объект «Отчет» для создания выходных документов в соответствии с требованиями конкретного пользователя.

Отчеты можно создавать в режиме автоотчета, с помощью мастера и в режиме конструктора. Технология создания отчетов очень похожа на технологию создания форм, т.к. многие операции выполняются аналогичным образом, однако имеются и дополнительные средства, позволяющие, например, выполнять группировку и сортировку данных в отчете, получать многостраничные отчеты и проставлять номера их страниц, создавать отчеты в виде этикеток и почтовых наклеек, получать снимки отчетов (файлы с расширением .snr), которые можно рассылать пользователям локальной сети, передавать по электронной почте или публиковать в Интернете.

### Структура отчета

Окно конструктора отчетов может включать в себя следующие разделы:



Заголовок и примечание отчета служат для отображения информации, размещаемой соответственно в начале первой и конце последней страницы. Эти разделы добавляются в отчет с помощью команды меню **Вид® Заголовок/примечание отчета**.

Верхний и нижний колонтитулы служат для размещения служебной информации, повторяющейся на каждой странице соответственно вверху и внизу. Эти разделы добавляются в отчет с помощью команды меню **Вид® Колонтитулы**.

Область данных служит для отображения всего множества строк источника записей, связанного с отчетом.

Если используется группирование данных отчета по какому-либо полю, то все строки источника записей разбиваются на группы, которые могут снабжаться заголовком и примечанием.

Для выполнения операции группирования данных нужно выполнить команду меню **Вид® Сортировка и группировка** (или нажать соответствующую кнопку на панели инструментов). В верхней строке появившегося окна необходимо указать поле, по которому будет выполняться группирование данных и порядок их сортировки. Для многоуровневой группировки нужно заполнить не одну, а несколько строк. Первая строка определяет основную группу, а последующие – подгруппы внутри групп. В нижней части окна указывается, нужно ли наделять группы данных областью заголовка и областью примечания, а также способ группирования: по полному значению поля либо по заданному числу первых символов для текстового поля и интервалу значений для числового поля или поля типа дата/время.

#### **Вычисления в отчетах**

Элемент управления «Поле», вставляемый в отчет, можно использовать для выполнения различных вычислений, если в его свойство «Данные» записать соответствующее выражение с использованием арифметических и логических операций, встроенных функций и констант. Простые формулы вводятся прямо в окне свойств, а для сложных выражений можно использовать построитель выражений. Например, итоговые данные по группе записей или по всему отчету можно подсчитать и вывести в области «Примечание группы» или «Примечание отчета» соответственно. Текущую дату или дату и время можно отобразить в области колонтитула или в области заголовка отчета с помощью встроенных функций Date() и Now() или же с помощью команды меню **Вставка® Дата и время**. Номер страницы в области колонтитула можно задать с помощью выражения

"Страница " & [Page] & " из " & [Pages],

где Page и Pages – свойства отчета, задающие номер текущей страницы и общее число страниц в отчете. Указанное выражение можно ввести вручную либо сформировать с помощью построителя выражений.

#### **Составные отчеты**

Так же как и в случае составных форм, составные отчеты используют данные из таблиц, связанных отношением «один ко многим», и конструируются аналогичным образом путем внедрения в область данных создаваемого отчета, или уже существующего отчета, или отчета, формируемого с помощью мастера на основе списка полей подчиненной таблицы.

## **Варианты заданий**

**Внимание!** В каждом отчете в области верхнего колонтитула нужно отобразить текущую дату и время, а также номер страницы.

1. Создайте отчет **«отч1: Данные запроса на выборку 1»**, предназначенный для отображения и вывода на печать значений следующих полей запроса «Запрос на выборку 1»: ГруппаОС, ИнвентарныйНомер, НаименованиеОС, Фамилия, НормаАмортизации, ПервоначСтоимость, СуммаИзноса. Данные в отчете должны быть сгруппированы по значениям поля «ГруппаОС» в порядке убывания. Для каждой группы и для всего отчета должны быть подсчитаны итоговые данные по числовым полям отчета. Отчет после его открытия должен выглядеть примерно так, как изображено на рис. 4.1.

2. Создайте отчет **«отч2: Данные запроса на выборку 2»**, предназначенный для отображения и вывода на печать значений следующих полей запроса «Запрос на выборку 2»: ИнвентарныйНомер, НаименованиеОС, Подразделение, Фамилия, ПервоначСтоимость, СодержаниеЗолота, СодержаниеСеребра, СодержаниеПлатины. Данные в отчете должны быть сгруппированы по значениям поля «Подразделение». Для каждой группы и для всего отчета должны быть подсчитаны итоговые данные по числовым полям отчета.

3. Создайте отчет **«отч3: Данные запроса на выборку 3»**, предназначенный для отображения и вывода на печать значений следующих полей запроса «Запрос на выборку 3»: ВидОС, ПодвидОС, ИнвентарныйНомер, НаименованиеОС, ПервоначСтоимость, НормаАмортизации, СуммаИзноса. Данные в отчете должны быть сгруппированы по значениям поля «ВидОС». Для каждой группы и для всего отчета должны быть подсчитаны итоговые данные по числовым полям отчета.

4. Создайте отчет **«отч4: Данные запроса параметрического 1»**, предназначенный для отображения и вывода на печать значений следующих полей запроса «Запрос параметрический 1»: ПодвидОС, ИнвентарныйНомер, НаименованиеОС, КодПодразделения, ПервоначСтоимость, НормаАмортизации, СуммаИзноса. Данные в отчете должны быть сгруппированы по значениям поля «ПодвидОС». Для каждой группы и для всего отчета должны быть подсчитаны итоговые данные по числовым полям отчета.

отч1: Данные запроса на выборку 1 : Report

09.12.2006 15:31:33 Страница 1 из 1

Инв.№	Наименование ОС:	Фамилия:	Норма аморти.	Первонач. стоим:	Сумма износа:
<b>Группа ОС: Промышленно-производственные</b>					
920	Фрезерный станок	Тихонов	5	27 368 000,00р.	1 614 290,00р.
2673	Здание гаража	Прошкина	1,2	26 400 000,00р.	220 800,00р.
180	Автомобиль АУДИ	Савельев	14,3	99 348 000,00р.	267 000,00р.
179	Автомобиль ВВЛТ 28	Прошкина	0,3	38 467 000,00р.	5 757 781,00р.
178	Автомобиль ГАЗ-53	Савельев	0,37	22 240 000,00р.	4 873 680,00р.
<b>Всего по группе ОС</b>				<b>213 823 000,00р.</b>	<b>12 733 551,00р.</b>
<b>Группа ОС: Производственные других отраслей</b>					
201	Мост ж/б арочный	Савельев	1	87 353 000,00р.	823 653,00р.
<b>Всего по группе ОС</b>				<b>87 353 000,00р.</b>	<b>823 653,00р.</b>
<b>Группа ОС: Непроизводственные</b>					
3270	Рабочая станция VAX-2201	Юрьева	12,5	95 287 600,00р.	11 934 539,00р.
3250	Компьютер NTT	Юрьева	10	41 755 400,00р.	288 140,00р.
<b>Всего по группе ОС</b>				<b>137 043 000,00р.</b>	<b>12 222 679,00р.</b>
<b>Итого по всем группам ОС</b>				<b>438 219 000,00р.</b>	<b>25 779 883,00р.</b>

Page: 1

Рис. 4.1. Отчет «отч1: Данные запроса на выборку 1»

5. Создайте отчет «отч5: Данные запроса параметрического 2», предназначенный для отображения и вывода на печать значений следующих полей запроса «Запрос параметрический 2»: Подразделение, КодМатОтв, ИнвентарныйНомер, НаименованиеОС, РежимИспользования, ПервоначСтоимость, СуммаИзноса. Данные в отчете должны быть сгруппированы по значениям поля «РежимИспользования». Для каждой группы и для всего отчета должны быть подсчитаны итоговые данные по числовым полям отчета.

6. Создайте отчет «отч6: Данные запроса с текстовым вычисляемым полем», предназначенный для отображения и вывода на печать значений следующих полей запроса «Запрос с текстовым вычисляемым полем»: ИнвентарныйНомер, НаименованиеОС, Подразделение, КодМатОтв, ФИО. Данные в отчете должны быть сгруппированы по значениям поля «Подразделение». Для каждой группы и для всего отчета должны быть подсчитаны количественные показатели (число строк).

7. Создайте отчет **«отч7: Данные запроса с числовым вычисляемым полем»**, предназначенный для отображения и вывода на печать значений следующих полей запроса «Запрос с числовым вычисляемым полем»: ИнвентарныйНомер, НаименованиеОС, КодНормыАморт, ПервоначСтоимость, СуммаИзноса, ОстаточнаяСтоимость, СрокАмортизации. Данные в отчете должны быть сгруппированы по значениям поля «КодНормыАморт». Для каждой группы и для всего отчета должны быть подсчитаны итоговые данные по числовым полям отчета.

8. Создайте отчет **«отч8: Данные запроса с группировкой»**, предназначенный для отображения и вывода на печать значений всех полей запроса «Запрос с группировкой». Данные в отчете, в свою очередь, должны быть сгруппированы по значениям поля «ГруппаОС» в порядке убывания. Для каждой группы и для всего отчета должны быть подсчитаны итоговые данные по числовым полям отчета.

9. Создайте отчет **«отч9: Данные запроса перекрестного»**, предназначенный для отображения и вывода на печать значений всех полей запроса «Запрос перекрестный». Данные в отчете, в свою очередь, должны быть сгруппированы по значениям поля «ГруппаОС» в порядке убывания. Для каждой группы и для всего отчета должны быть подсчитаны итоговые данные по числовым полям отчета.

10. Создайте отчет **«отч10: Пробег автотранспортных ОС»**, предназначенный для отображения и вывода на печать значений следующих полей базы данных: ПодвидОС, ИнвентарныйНомер, НаименованиеОС, Месяц, Год, Пробег\_км. Данные в отчете должны быть сгруппированы по годам, а внутри каждого года – по месяцам. Для каждой группы (год), подгруппы (месяц) и для всего отчета должен быть подсчитан суммарный пробег в километрах. В результате отчет должен выглядеть примерно так, как показано на рис. 4.2.

11. Создайте отчет **«отч11: Составной отчет 1»**, предназначенный для отображения и вывода на печать значений всех полей таблицы «Нормы амортизации». Этот отчет должен содержать в себе подчиненный отчет, созданный на основе существующей формы «фрм11: Инвентарные карточки ОС».

12. Создайте в режиме мастера отчет **«отч12: Первоначальная стоимость ОС»** в виде диаграммы, отображающей итоговую первоначальную стоимость основных средств в разрезе кодов подразделений и кодов материально ответственных. Для этого предварительно создайте запрос, содержащий поля «КодПодразделения», «КодМатОтв», «ПервоначСтоимость». В результате отчет должен выглядеть примерно так, как показано на рис. 4.3.

Microsoft Access - [отч10: Пробег автотранспортных ОС : Report]

## Пробег автотранспортных ОС

29.08.2006 20:14:15 Страница 1 из 1

Подвид ОС:	Инв.№:	НаименованиеОС:	Пробег_км:
<b>Год: 2004</b>			
<b>Месяц: 12</b>			
Автомобили(более 2 т) с ресурсом от 200 до 250 тыс.	179	Автомобиль VWЛТ28	167
Автомобили(более 2 т) с ресурсом до 200 тыс.км	178	Автомобиль ГАЗ-53	134
<b>Всего за месяц</b>			<b>301</b>
<b>Всего за год</b>			<b>301</b>
<b>Год: 2005</b>			
<b>Месяц: 2</b>			
Автомобили(более 2 т) с ресурсом до 200 тыс. км	178	Автомобиль ГАЗ-53	456
<b>Всего за месяц</b>			<b>456</b>
<b>Месяц: 3</b>			
Автомобили(более 2 т) с ресурсом от 200 до 250 тыс.	179	Автомобиль VWЛТ28	57
<b>Всего за месяц</b>			<b>57</b>
<b>Всего за год</b>			<b>513</b>
<b>Год: 2006</b>			
<b>Месяц: 3</b>			
Автомобили(более 2 т) с ресурсом до 200 тыс. км	178	Автомобиль ГАЗ-53	529
<b>Всего за месяц</b>			<b>529</b>
<b>Месяц: 4</b>			
Автомобили(более 2 т) с ресурсом от 200 до 250 тыс.	179	Автомобиль VWЛТ28	456
Автомобили(более 2 т) с ресурсом до 200 тыс. км	178	Автомобиль ГАЗ-53	1128
<b>Всего за месяц</b>			<b>1584</b>
<b>Месяц: 5</b>			
Автомобили(более 2 т) с ресурсом от 200 до 250 тыс.	179	Автомобиль VWЛТ28	92
<b>Всего за месяц</b>			<b>92</b>
<b>Всего за год</b>			<b>2205</b>
<b>Итого</b>			<b>3019</b>

Page: 1

Рис. 4.2. Отчет «отч10: Пробег автотранспортных ОС»

13. Создайте в режиме мастера отчет «отч13: Суммарный износ по группам и видам ОС» в виде диаграммы, отображающей суммы износа основных средств в разрезе кодов групп ОС и кодов видов ОС. Для этого предварительно создайте запрос, содержащий поля «КодГруппыОС», «КодВидаОС», «СуммаИзноса». В результате отчет должен выглядеть примерно так, как показано на рис. 4.4.

14. Создайте отчет «отч14: Составной отчет 2», предназначенный для отображения и вывода на печать всех значений полей таблицы «Нормы амортизации». Этот отчет должен содержать в себе подчиненный отчет, отображаю-

щий значения следующих полей таблицы «Инвентарные карточки ОС»: ИнвентарныйНомер, НаименованиеОС, КодМатОтв, ПервоначСтоимость, СуммаИз-носа.



Рис. 4.3. Отчет «отч12: Первоначальная стоимость ОС»



Рис. 4.4. Отчет «отч13: Суммарный износ по группам и видам ОС»

15. Создайте отчет «**отч15: Составной отчет 3**», предназначенный для отображения и вывода на печать значений следующих полей таблиц базы данных: КодПодразделения, Подразделение, КодМатОтв, Фамилия, Имя, Отчество. Этот отчет должен содержать в себе подчиненный отчет, отображающий значения следующих полей таблицы «Инвентарные карточки ОС»: ИнвентарныйНомер, НаименованиеОС, ДатаПоступления, ПервоначСтоимость, СуммаИзноса.

Библиотека БГУИР

## Лабораторная работа №5. Создание макросов

### Цель работы

1. Ознакомиться с процессом создания макросов.
2. Освоить способы формирования условий в макросах и способы формирования групп макросов.
3. Ознакомиться со способами использования макросов в качестве обработчиков событий в формах и отчетах.

### Задачи

1. Создание макроса.
2. Формирование условий, определяющих порядок выполнения макрокоманд.
3. Создание вспомогательной формы, использующей макрос в качестве обработчика конкретных событий.
4. При необходимости (зависит от варианта задания) создание группы макросов.

### Методические указания

Макросом называется набор из одной или нескольких макрокоманд, представляющих собой заранее созданные программы для выполнения некоторых часто используемых операций, например, открытия форм, выполнения запросов, печати отчетов, применения фильтров, передачи фокуса ввода от одного элемента управления к другому, присвоения значений и их передачи через буфер обмена, выполнения и создания команд меню, выполнения разнообразных команд и т.д.

Макросы можно запускать из окна базы данных, а также с помощью кнопок, размещаемых на формах и панелях инструментов, однако наиболее часто макросы используются в качестве обработчиков событий, возникающих при работе с базой данных. Например, для элемента управления «Поле» можно создать макросы для обработки событий, возникающих в следующих случаях:

- до и после операции обновления значения поля;
- при любом изменении значения поля;
- при получении и потере фокуса ввода;
- при использовании мыши: щелчок, двойной щелчок, нажатие кнопки вниз, нажатие кнопки вверх;
- при использовании клавиатурных клавиш: нажатие клавиши, движение клавиши вниз, движение клавиши вверх.

Макросы создаются в режиме конструктора. Каждая строка в верхней части окна конструктора предназначена для задания конкретной макрокоманды, выбираемой из выпадающего списка в столбце «Макрокоманда». Справа, в столб-

це «Примечание», можно при желании поместить комментарий к данной макрокоманде. Если макрос содержит несколько макрокоманд, то необходимо заполнить несколько строк.

В нижней части окна конструктора для каждой макрокоманды указываются ее аргументы, число и состав которых зависят от конкретной макрокоманды.

Для проверки работы созданного макроса при открытом окне конструктора нужно нажать кнопку **Запуск** на панели инструментов (или использовать команду меню **Запуск® Запуск**). При этом будут выполнены все макрокоманды, входящие в макрос, в том порядке, в котором они перечислены в окне конструктора. Для отладки сложных макросов можно использовать пошаговый режим, в котором каждый раз выполняется только одна макрокоманда после чего выполнение макроса приостанавливается. Затем можно выполнить следующую макрокоманду и т.д. Для перехода в пошаговый режим необходимо нажать кнопку **По шагам** панели инструментов (или воспользоваться командой меню **Запуск® По шагам**).

### **Макросы с условиями**

В некоторых случаях требуется выполнить одну или несколько макрокоманд только при выполнении определенного условия. Условие задается выражением логического типа, которое может включать в себя арифметические и логические операторы, константы, встроенные функции, имена полей, элементов управления и свойств. Условие записывается слева от имени макрокоманды в столбце «Условие», который добавляется в окно конструктора нажатием кнопки **Условия** на панели инструментов.

Если условие истинно, выполняется макрокоманда, находящаяся в той же строке, а также все идущие подряд макрокоманды, у которых в ячейках столбца «Условие» содержится многоточие (...). После этого выполняются макрокоманды, у которых ячейки столбца «Условие» являются пустыми.

Если условие ложно, то текущая макрокоманда игнорируется вместе со всеми следующими за ней макрокомандами, у которых в ячейках столбца «Условие» содержится многоточие, и совершается переход к ближайшей строке, в которой ячейка столбца «Условие» не заполнена.

### **Группы макросов**

При большом числе макросов их часто разбивают на группы, обычно родственные по своему назначению. Каждая группа создается в одном и том же окне конструктора и сохраняется под общим именем. Макросы внутри группы также снабжаются собственными именами, задаваемыми в ячейках столбца «Имя макроса». Этот столбец добавляется в окно конструктора с помощью кнопки **Имена макросов** на панели инструментов. Поэтому имя макроса, входящего в группу макросов, является составным и состоит из имени группы и имени макроса, разделяемых точкой.

## Варианты заданий

1. Создайте вспомогательную форму «фрм: для макроса 1», предназначенную для ввода и редактирования данных в таблицу «Пробег автотранспортных ОС». Далее создайте макрос «**Макрос 1**», обеспечивающий следующие возможности: если после открытия существующей формы «фрм1: Просмотр запроса на выборку 1» начать перемещение по отображаемым ею записям с помощью кнопок навигации, то каждый раз при появлении автотранспортного ОС (с установленным флажком «Учитывать пробег») должна открываться вспомогательная форма в режиме таблицы, причем эта таблица должна содержать только те данные, которые относятся к текущему автотранспортному ОС. В противном случае на экран должно выдаваться сообщение, что основное средство не является автотранспортным.

2. Создайте группу «**Группа макросов 2**» из макросов, обеспечивающих следующие возможности. После открытия существующей формы «фрм2: Просмотр запроса на выборку 2» она должна быть прижата к верхнему и левому краям рабочего окна Access. Если выполнить двойной щелчок мышью в одном из полей «ИнвентарныйНомер», «КодПодразделения», «КодМатОтв» этой формы, то должна открыться одна из таблиц «Инвентарные карточки ОС», «Подразделения предприятия», «Материально ответственные» соответственно, причем каждая из них должна автоматически закрыться не более чем через 10 секунд после открытия. Двойной щелчок мышью на свободном месте формы должен открывать не более чем на 10 секунд запрос «Запрос на выборку 2» в режиме конструктора. Если закрыть форму «фрм2: Просмотр запроса на выборку 2», а какие-либо из таблиц или запрос еще не закрылись, то они также должны сразу закрыться.

3. Создайте вспомогательную форму «фрм: для группы макросов 3», содержащую 6 кнопок, расположенных в два ряда (2 кнопки в первом ряду и 4 – во втором). Нажатие на первую и вторую кнопки в первом ряду должно соответственно открывать и закрывать уже существующую форму «фрм3: Просмотр запроса на выборку 3» (при ее открытии должна быть полностью видна и вспомогательная форма). Четыре кнопки во втором ряду предназначены для навигации по набору данных, отображаемых посредством формы «фрм3: Просмотр запроса на выборку 3» (позиционирование на первую, предыдущую, следующую и последнюю строки соответственно). При этом открытие вспомогательной формы и функции, выполняемые ее кнопками, необходимо реализовать с помощью группы макросов с общим именем «**Группа макросов 3**».

*Усложненный вариант.* Устраните появление на экране сообщений, сигнализирующих о попытке переместиться на несуществующую строку (т.е. на строку перед первой или после последней).

4. Создайте макрос «**Макрос 4**», реализующий следующие возможности. Если открыть уже существующую форму «фрм4: Подвиды ОС», выбрать в списке конкретный подвид ОС и нажать кнопку открытия вспомогательной формы,

то должен быть проведен анализ: имеются или отсутствуют инвентарные карточки, отображаемые посредством этой формы. Если таких инвентарных карточек нет и вспомогательная форма отображает поля новой, пустой записи, то на экран нужно выдать соответствующее сообщение, после чего некоторые поля пустой записи следует заполнить данными. В частности, в поле «КодНормыАморт» нужно занести значение, соответствующее выбранному подвиду ОС, в поле «РежимИспользования» – значение «В эксплуатации», в поле «ДатаПоступления» – текущую дату.

5. Создайте форму на основе запроса «Параметрический запрос 2» и далее реализуйте с помощью макроса «**Макрос 5**» операцию открытия этой формы и последующий анализ набора данных, отображаемых посредством формы. Если набор данных содержит хотя бы одну запись, то на экран должно быть выдано соответствующее сообщение. Если набор данных пуст, то вспомогательная форма должна закрыться и на экран должно быть выведено сообщение, что записи отсутствуют. В этом случае после закрытия пользователем окна с сообщением должна открыться таблица «Инвентарные карточки ОС» с отображением всех записей и позиционированием на строке ввода новой записи.

6. Создайте вспомогательную форму «фрм: для макроса 6» для ввода и редактирования данных в таблицу «Инвентарные карточки ОС», располагая поля ввода в столбец. Далее создайте группу «**Группа макросов 6**» из трех макросов. Первый из них должен открывать сначала вспомогательную форму, а затем существующую форму «фрм6: Просмотр запроса с текстовым вычисляемым полем». Второй макрос должен обеспечивать следующее: если в форме «фрм6: Просмотр запроса с текстовым вычисляемым полем» перемещаться по записям с помощью кнопок навигации, то вспомогательная форма должна автоматически позиционироваться так, чтобы значения полей «КодМатОтв» в обеих формах были равны. Третий макрос при закрытии одной формы должен обеспечить автоматическое закрытие другой формы.

7. Создайте вспомогательную форму «фрм: для макроса 7» для ввода и редактирования данных в таблицу «Материально ответственные», располагая поля ввода в столбец. В существующую форму «фрм7: Просмотр запроса с числовым вычисляемым полем» вставьте элемент управления типа «кнопка». Далее создайте макрос «**Макрос 7**», который должен обеспечить выполнение следующих действий. Если открыть форму «фрм7: Просмотр запроса с числовым вычисляемым полем», а затем переместиться на какую-либо запись и нажать упомянутую выше кнопку, то эта форма должна закрыться, после чего должна открыться вспомогательная форма в режиме добавления новой записи, причем в ее поле «КодПодразделения» должно быть установлено значение, которое сохранилось в аналогичном поле закрывшейся формы.

8. Создайте параметрический запрос на выборку «Запрос для макроса 8», выводящий значения полей инвентарной карточки по указанному пользователем инвентарному номеру, после чего создайте форму «фрм: для макроса 8» на основе этого запроса. Далее создайте макрос «**Макрос 8**», открывающий форму «фрм: для макроса 8» с табличным представлением данных. Если же инвентар-

ной карточки с указанным номером не существует, то макрос должен закрыть форму «фрм: для макроса 8» и выдать на экран соответствующее сообщение. В этом случае после закрытия пользователем окна с сообщением должна открыться существующая форма «фрм8: Инвентарные карточки ОС» в режиме ввода новой записи, причем поле «РежимИспользования» уже должно содержать значение «Сдано в аренду», а поле «ДатаВыбытия» – текущую дату.

9. Создайте форму «фрм: для макроса 9», содержащую элемент управления типа «поле» со списком для просмотра всех имеющихся подвидов основных средств и выбора одного из них. Создайте макрос «**Макрос 9**», который обеспечивает следующую возможность. Как только из списка будет выбран конкретный подвид ОС, то сразу же должна открыться уже существующая форма «фрм9: Ленточные инвентарные карточки», причем она должна отображать только те инвентарные карточки, код нормы амортизации которых совпадает с кодом нормы амортизации выбранного в списке подвида ОС. Если такие инвентарные карточки отсутствуют, то форма «фрм9: Ленточные инвентарные карточки» на экране не отображается, а вместо нее выводится сообщение об отсутствии инвентарных карточек.

10. Создайте группу «**Группа макросов 10**» из четырех макросов, первый из которых открывает уже существующую основную форму «фрм10: Виды – Нормы амортизации», а остальные реализуют следующие возможности. Если выполнить двойной щелчок мышью в верхней или нижней части основной формы, то должна открыться вспомогательная, также существующая форма «фрм10: Табличные нормы амортизации». Если же выполнить двойной щелчок мышью в поле ввода «КодВидаОС» основной формы, то во вспомогательной форме должны отобразиться только те записи, для которых значение поля «КодВидаОС» совпадает с аналогичным значением в основной форме. При закрытии основной формы должна закрываться также и вспомогательная форма (если, конечно, она была перед этим открыта).

11. Создайте вспомогательную форму «фрм: для макроса 11», содержащую поле ввода и кнопку. Нажатие на кнопку должно запускать макрос «**Макрос 11**», который открывает существующую форму «фрм11: Инвентарные карточки ОС» и позиционирует источник записей на строке, для которой значение поля «ДатаПоступления» совпадает со значением, установленным в поле ввода вспомогательной формы. Если же инвентарная карточка с такой датой поступления отсутствует, то форма «фрм11: Инвентарные карточки ОС» должна закрыться и на экран должно быть выведено соответствующее сообщение.

12. Создайте вспомогательную форму «фрм: для макроса 12», содержащую поле ввода и кнопку. Нажатие на кнопку должно запускать макрос «**Макрос 12**», который открывает существующую форму «фрм12: Инвентарные карточки ОС» в табличном виде, причем в ней должны отображаться только те записи, для которых значение поля «КодГруппыОС» совпадает со значением, содержащимся в поле ввода вспомогательной формы. Если такие записи отсутствуют, то на экран должно быть выведено соответствующее сообщение.

13. Создайте группу «**Группа макросов 13**» из трех макросов, первый из которых открывает существующую форму «фрм13: Инвентарные карточки ОС», а другие два обеспечивают следующие возможности. Если в указанной выше форме нажать кнопку **Открыть**, то должна открыться вспомогательная форма «фрм13: Материально ответственные» с позиционированием на новой, пустой записи. Если же сделать двойной щелчок мышью в области данных вспомогательной формы, то должно выполниться позиционирование на записи, в которой значение поля «КодМатОтв» совпадает с аналогичным значением в форме «фрм13: Инвентарные карточки ОС».

14. Создайте форму «фрм: для группы макросов 14», содержащую 5 кнопок. Первая кнопка должна открывать все имеющиеся таблицы базы данных. Вторая и третья кнопки должны создавать на экране соответственно горизонтальную и вертикальную мозаику из окон. Четвертая кнопка должна располагать окна на экране каскадом. Пятая кнопка должна закрывать все открытые таблицы базы данных. Для выполнения всех этих операций создайте группу макросов с общим именем «**Группа макросов 14**».

*Усложненный вариант.* Добавьте в форму еще две кнопки, одна из которых сворачивает (минимизирует), а вторая – восстанавливает (в прежней позиции и размерах) окна всех открытых таблиц базы данных.

15. Создайте вспомогательные (одиночную и табличную) формы «фрм: для макроса 15 одиночная» и «фрм: для макроса 15 табличная», которые обе предназначены для ввода и редактирования данных в таблицу «Инвентарные карточки ОС» (табличный тип формы устанавливается с помощью окна свойств выбором вкладки «Макет» и строки «Режим по умолчанию»). Снабдите одиночную форму кнопкой с надписью **Копировать**. Далее создайте макрос «**Макрос 15**», обеспечивающий следующие возможности: если в открытой одиночной форме с помощью кнопок навигации выбрать некоторую запись, после чего нажать кнопку **Копировать**, то в табличной форме должна быть создана копия записи, выбранной в одиночной форме, т.е. добавлена новая запись с идентичными значениями всех полей за исключением ключевого поля «ИнвентарныйНомер», которое должно заполняться вручную. Если при нажатии кнопки **Копировать** табличная форма была закрыта, то ее нужно открыть и только затем выполнить описанную выше операцию копирования.

## Лабораторная работа №6. Создание приложений

### Цель работы

Ознакомиться с процессом разработки кнопочных форм и пользовательских панелей инструментов.

### Задачи

1. Создание главной кнопочной формы и настройка ее элементов управления.
2. Настройка параметров запуска базы данных.
3. Создание пользовательской панели инструментов.

### Методические указания

Создание приложения, представляющего собой одну или несколько кнопочных форм, можно считать заключительным (но не обязательным) этапом разработки базы данных, поскольку этот этап предназначен всего лишь для упрощения работы пользователя с базой данных. Обычно одна из кнопочных форм является главной и служит чем-то вроде титульного листа и меню приложения, позволяя с помощью размещенных на форме кнопок открывать отчеты и формы, выполнять запросы и макросы, печатать отчеты и др.

Для создания приложения нужно выбрать команду меню **Сервис® Служебные программы® Диспетчер кнопочных форм**. Далее в появившемся окне «Диспетчер кнопочных форм» необходимо нажать кнопку **Изменить**. Появляется окно «Изменение страницы кнопочной формы», в котором собственно и создаются кнопки кнопочной формы, указывается их назначение и команды, выполняемые при нажатии. При необходимости в окне «Диспетчер кнопочных форм» можно создать несколько кнопочных форм, после чего каждую из них снабдить своим набором кнопок, в том числе кнопками, позволяющими открывать другие кнопочные формы. Кнопочные формы можно создавать и обычным способом с помощью конструктора форм и панели элементов, однако этот способ более трудоемкий.

Главная кнопочная форма может быть выведена на экран при открытии базы данных вместо окна базы данных. Для этого, а также для изменения стандартного вида экрана нужно выполнить настройку параметров запуска базы данных с помощью команды меню **Сервис® Параметры запуска**. В результате будет создано приложение для работы с базой данных на основе кнопочных форм, предоставляющих пользователю лишь те возможности, которые ему необходимы с точки зрения разработчика базы данных и кнопочных форм.

## Задание к работе

1. Создайте главную кнопочную форму с шестью рядами кнопок: кнопки для открытия всех таблиц, кнопки для открытия всех имеющихся запросов, кнопки для открытия всех имеющихся форм, кнопки для открытия всех имеющихся отчетов и кнопки для запуска всех имеющихся макросов. Опробуйте работу этой формы.

2. Настройте параметры запуска главной кнопочной формы с помощью команды меню **Сервис® Параметры запуска**. Из раскрывающегося списка «Вывод формы/страницы» выберите имя главной кнопочной формы. Для исключения вывода на экран окна базы данных при ее запуске уберите флажок «Окно базы данных». Закройте окно базы данных, а затем снова откройте базу данных. Вместо окна базы данных вы должны увидеть на экране главную кнопочную форму.

3. Создайте пользовательскую панель инструментов, с помощью которой можно открыть одну таблицу (например «Инвентарные карточки ОС»), один запрос, одну форму, один отчет и запустить один макрос. Расположите эту панель в крайнем нижнем положении.

Для создания такой панели инструментов выполните команду меню **Вид® Панели инструментов® Настройка** и в появившемся окне «Настройка» нажмите кнопку **Создать**. Затем укажите имя создаваемой панели инструментов и далее разместите на ней нужные кнопки путем перетаскивания соответствующих объектов базы данных.

## Литература

1. Дубнов, П. Ю. Access 2002 / П. Ю. Дубнов. – СПб. : ДМК Пресс, 2003.
2. Хомоненко, А. Д. Microsoft Access. Экспресс-курс / А. Д. Хомоненко, В. В. Гридин. – СПб. : БХВ-Петербург, 2005.
3. Сеннов, А. Access 2003. Практическая разработка баз данных : учебный курс / А. Сеннов. – СПб. : Питер, 2005.
4. Слепцова, Л. Д. Программирование на VBA : самоучитель / Л. Д. Слепцова. – М. : Издательский дом «Вильямс», 2004.
5. Автоматизированные информационные технологии в экономике : учеб. для вузов / В. В. Брага [и др.] ; под ред. Г. А. Титоренко. – М. : ЮНИТИ, 2002.
6. Артре, Ш. Структурный подход к организации баз данных / Ш. Артре ; пер. с англ. – М. : Финансы и статистика, 1989.
7. Хомоненко, А. Д. Базы данных : учебник для вузов / А. Д. Хомоненко, В. М. Цыганков, М. Г. Мальцев ; под ред. А. Д. Хомоненко. – СПб. : КОРОНА принт, 2004.
8. Практическое руководство по SQL / Д. Боуман [и др.]. – Киев, М. : Диалектика, 1997.
9. Грабор, М. Введение в SQL / М. Грабор. – М. : Лори, 1996.
10. Грофф, Дж. Р. Энциклопедия SQL / Дж. Р. Грофф, П. Н. Вайнберг. – СПб. : Питер, 2003.
11. Дейт, К. Дж. Введение в системы баз данных : учеб. пособие / К. Дж. Дейт. – М. : Издательский дом «Вильямс», 2001.
12. Змитрович, А. И. Базы данных : учеб. пособ. для вузов / А. И. Змитрович. – Минск : Университетское, 1991.
13. Карпова, Т. С. Базы данных : модели, разработка, реализация : учебник для вузов / Т. С. Карпова. – СПб. : Питер, 2001.
14. Когаловский, М. Р. Энциклопедия технологий баз данных / М. Р. Когаловский. – М. : Финансы и статистика, 2002.
15. Коннолли, Т. Базы данных / Т. Коннолли, К. Бегг. – М. : Издательский дом «Вильямс», 2003.
16. Кренке, К. Теория и практика построения баз данных / К. Кренке. – СПб. : Питер, 2005.
17. Малыхина, М. П. Базы данных : учеб. пособие / М. П. Малыхина. – СПб. : БХВ-Петербург, 2004.
18. Мартин, Д. Организация баз данных в вычислительных системах / Д. Мартин. – М. : Мир, 1980.
19. Мейер, Д. Теория реляционных баз данных / Д. Мейер. – М. : Мир, 1987.
20. Миронов, Д. А. Создание Web-страниц в MS Office 2000 : учеб. пособие / Д. А. Миронов. – СПб. : БХВ-Петербург, 2000.
21. Роб, П. Системы баз данных: проектирование, реализация и управление / П. Роб, К. Коронел. – СПб. : БХВ-Петербург, 2004.

22. Роб, П. Системы баз данных / П. Роб. – СПб. : БХВ-Петербург, 2004.
23. Роланд, Ф. Д. Основные концепции баз данных / Ф. Д. Роланд. – М. : Издательский дом «Вильямс», 2002.
24. Сичкаренко, В. А. SQL-99. Руководство разработчика баз данных / В. А. Сичкаренко. – СПб. : ООО «ДиаСофтЮП», 2002.
25. Хансен, Г. Базы данных: разработка и использование / Г. Хансен, Дж. Хансен ; пер. с англ. – М. : БИНОМ, 2000.

Библиотека БГУИР

Учебное издание

**Бутов Алексей Александрович**

**ТЕХНОЛОГИИ ОРГАНИЗАЦИИ, ХРАНЕНИЯ И ОБРАБОТКИ ДАННЫХ**

Лабораторный практикум  
для студентов экономических специальностей БГУИР  
всех форм обучения

Редактор М. В. Тезина  
Корректор Е. Н. Батурчик

---

Подписано в печать 30.05.2008.	Формат 60×84 1/16.	Бумага офсетная.
Гарнитура «Таймс».	Печать ризографическая.	Усл. печ. л. 3,14.
Уч.-изд. л. 3,1.	Тираж 150 экз.	Заказ 620.

---

Издатель и полиграфическое исполнение: Учреждение образования  
«Белорусский государственный университет информатики и радиоэлектроники»  
ЛИ №02330/0056964 от 01.04.2004. ЛП №02330/0131666 от 30.04.2004.  
220013, Минск, П. Бровки, 6