

Составитель Л.П. Андреева

Редактор Е.П. Храмова

Методические указания содержат описания лабораторных работ, справочный материал по работе с системой управления базами данных Access и список литературы. Материал предназначен для студентов, изучающих курс «Базы данных».

Печатаются по решению редакционно-издательского совета Московского государственного института радиотехники, электроники и автоматики (технического университета).

Рецензенты: Ю.Е. Мороховец, Б.М. Басок

© Московский государственный институт радиотехники, электроники и автоматики (технический университет), 2000

Литературный редактор О.В. Волкова

Изд. лиц. № 020456 от 04.03.97

Подписано в печать 01.03.2000. Формат 60 x 84 1/16.

Бумага офсетная. Печать офсетная. Усл. печ. л. 1,86

Усл.кр.-отт. 7,44. Уч.-изд. л. 2,0. Тираж 300 экз.

Заказ 371. Бесплатно

Московский государственный институт радиотехники, электроники и автоматики (технический университет)

117454, Москва, просп. Вернадского, 78

ЛАБОРАТОРНАЯ РАБОТА №1

КОНЦЕПТУАЛЬНАЯ МОДЕЛЬ ДАННЫХ

Создать концептуальную модель данных для предметной области. Варианты описания предметной области приведены в Приложении 1.

Методические указания

- Определите сущности предметной области.
- Для каждой сущности определите атрибуты и их характеристики. Характеристики атрибутов представьте в виде таблицы (табл. 1)

Таблица 1
Характеристики атрибутов сущности <имя сущности>

Имя атрибута	Тип	Допустимое значение	Размер	Обязательность	Значение по умолчанию

- Установите связи между сущностями и отобразите их в виде ER-диаграммы
- Преобразуйте ER-диаграмму в реляционную модель базы данных. Определите первичные ключи отношений.
- Нормализуйте базу данных.

ЛАБОРАТОРНАЯ РАБОТА №2

ПРОЕКТИРОВАНИЕ ТАБЛИЦ В СУБД ACCESS

Спроектировать таблицы базы данных «Служба занятости» и установить связи между ними. Списки полей проектируемых таблиц приведены в табл. 2 - 7.

Таблица 2
Список полей таблицы административного деления (Округ)

Имя поля	Тип	Размер	Обязательность	Маска ввода
Код Мо	Целый		Да	
Наименование Мо	Текст	20	Да	1 символ прописная буква
Ао	Текст	3	Да	Прописные буквы

Список полей таблицы работников (Fio)

Таблица 3

Имя поля	Тип	Размер	Обязательность	Маска ввода
Номер работника	Счетчик		Да	
Фамилия	Текст	24	Да	1 символ прописная буква
Имя	Текст	12	Да	1 символ прописная буква
Отчество	Текст	14	Да	1 символ прописная буква
Дата рождения	Дата		Да	
Адрес	Текст	40		
Код Мо	Целый			
Дата регистрации	Дата			

Список полей таблицы предприятий (Company) Таблица 4

Имя поля	Тип	Размер	Обязательность	Маска ввода
Код предприятия	Целый		Да	
Наименование предприятия	Текст	20	Да	1 символ прописная буква
Адрес	Текст	40		
Код Мо	Целый			

Список полей таблицы вакансий (Vacan)

Таблица 5

Имя поля	Тип	Размер	Обязательность	Маска ввода
Код предприятия	Целый		Да	
Код профессии	Целый		Да	
Опыт работы	Логический		Да	
Зарплата	Денежный		Да	
Количество вакансий	Целый		Да	

Список полей таблицы профессий (Prof)

Таблица 6

Имя поля	Тип	Размер	Обязательность	Маска ввода
Код профессии	Целый		Да	
Наименование профессии	Текст	20	Да	1 символ прописная буква

Список полей таблицы предложений (Work)

Таблица 7

Имя поля	Тип	Размер	Обязательность	Маска ввода
Номер работника	Целый		Да	
Код профессии	Целый		Да	
Опыт работы	Логический		Да	

Методические указания

При первом запуске Access создайте файл с базой данных.

Для создания структуры таблицы выполните следующие действия.

- Выберите в окне *База данных* закладку *Таблицы*.
- Нажмите кнопку *Создать*.
- Для каждого поля введите имя поля, тип, описание, свойства. Для задания значения свойству *Маска ввода* используйте символы, приведенные в табл. 8.

Символы для задания маски ввода

Таблица 8

Символ	Описание
0	Цифра, обязательность ввода
9	Цифра, необязательность ввода
#	Цифра, пробел, +, -, необязательность ввода
L	Буква A-Z, A-Я, обязательность ввода
?	Буква A-Z, A-Я, необязательность ввода
^	Буква или цифра, обязательность ввода
a	Буква или цифра, необязательность ввода
&	Любой символ или пробел, обязательность ввода
C	Любой символ или пробел, необязательность ввода
<	Перевод следующих символов в нижний регистр
>	Перевод следующих символов в верхний регистр
!	Ввод символов справа налево

- Установите первичные ключи таблицы.
- Сохраните таблицу и закройте окно *Таблица*.
Для создания связей между таблицами выполните следующие действия.
- Выберите в окне *База данных* пиктограмму *Схема данных*.
- Добавьте по очереди таблицы в окно *Схема данных*.
- Протяните связи между полями таблиц, заполняя для каждой связи окно *Связь*.
- Сохраните связи и закройте окно *Схема данных*.

ЛАБОРАТОРНАЯ РАБОТА №3

РАБОТА С ТАБЛИЦАМИ

Ввести данные в таблицы

Список муниципальных округов

101, Таганский, ЦО

103, Тверской, ЦО

905, Олимпийская деревня, ЮЗО

906, Ясенево, ЮЗО

Список профессий

11, Программист

12, Слесарь

13, Бухгалтер

14, Менеджер

15, Инженер

Список работников

1, Иванов Карл Петрович, 01.03.43, 117488, Песочная 33-105, 103, текущая дата

2, Петров Сергей Николаевич, 05.02.55, 117287, Живописная 3-25, 905, текущая дата

3, Васильев Петр Иванович, 09.03.70, 117463, Голубинская 32-48, 906, текущая дата

4, Андреев Олег Александрович, 11.02.70, 117463, Тарусская 8-150, 906, текущая дата

Список предложений работников

1, 11, Нет

1, 12, Да

2, 12, Нет

3, 11, Да

4, 13, Нет

4, 15, Да

Список предприятий

101, Хлебокомбинат, 117221, Заводская 1, 101

103, Банк, 117223, Владимирская 24, 103

105, Магазин, 117221, Заводская 4, 101

107, Автобаза, 117287, Озерная 5, 905

Список вакансий

103, 11, Нет, 1000, 3

103, 12, Нет, 3000, 1

101, 15, Нет, 2000, 2

101, 11, Да, 6000, 2

107, 13, Да, 10000, 1

105, 11, Нет, 2000, 1

105, 14, Да, 2000, 1

Изменить структуру таблицы *Гео*, добавив в нее новое поле *Телефон*.
Ввести значения в новое поле для каждой записи.

Установить для поля *Дата рождения* условие на значение $> \#01/01/1930\#$. Установить в свойстве *Сообщение об ошибке* этого поля значение «Ошибка ввода даты».

ЛАБОРАТОРНАЯ РАБОТА №4

ЗАПРОСЫ QBE

1. Запросы выбора

1.1. Однотабличные запросы

- Вывести все записи таблицы *Okrug*, отсортировав их по *Ао*, а внутри *Ао* по наименованию муниципальных округов.
- Вывести наименования и адреса всех предприятий, расположенных в 101 или 103 округах.
- Вывести фамилии, имена и телефоны работников, проживающих в 103 и зарегистрированных в службе занятости в текущем году.
- Вывести наименования всех муниципальных округов, в наименовании которых вторая буква «а».

1.2. Многотабличные запросы

- Найти все предприятия, для которых требуются программисты без опыта работы, указав для них количество вакансий и зарплату.
- Найти все предприятия, для которых требуются программисты без опыта работы или бухгалтера с опытом работы.

1.3. Итоговые запросы с группированием

- Найти количество работников, зарегистрированных в службе занятости, для каждого Ао. Отсортировать записи выходного набора по Ао.
- Найти количество вакансий, не требующих опыта работы по каждой профессии, указав в выходном наборе наименование профессии и количество вакансий. Установить заголовок второго столбца в «Сумма вакансий».
- Найти общее количество вакансий службы занятости.
- Найти среднюю, минимальную и максимальную зарплаты для каждого предприятия.

1.4. Использование вычисляемых полей в запросах

- Вывести для всех работников, проживающих в ЦО, года рождений. Результат вывести в виде таблицы, изображенной на рис. 1.

Работник	Год рождения
Андреев Иван	1970
Иванов Карл	1943

Рис. 1. Выходной набор запроса с вычисляемым полем *Год рождения*

- Найти среднюю зарплату, предлагаемую службой занятости работникам, не имеющим опыта работы.
- Найти среднюю зарплату, предлагаемую работникам, не имеющим опыта работы, по каждой профессии. Отсортировать выводимые записи по полю *Профессия*. Результат вывести в виде таблицы, изображенной на рис. 2.

Профессия	Средняя зарплата
Бухгалтер	10000,00 р.
Инженер	2000,00 р.

Рис. 2. Выходной набор запроса с вычисляемым полем *Средняя зарплата*

2. Параметрические запросы

- Найти список предприятий, на которых есть вакансии для указанной пользователем профессии.
- Найти все вакансии для работника, используя в качестве параметров запроса его фамилию и имя. Указать в выходном наборе наименования и адреса предприятий, наименования профессий, зарплаты. Сохранить запрос под именем «Вакансии!».

- Найти всех кандидатов на указанную профессию указанного предприятия. Указать в выходном наборе фамилию, имя, отчество и телефон для каждого кандидата. Сохранить запрос под именем «Работники!».

3. Запросы действия

- 3.1. Запрос на обновление группы записей
 - Увеличить всем программистам банка зарплату в два раза.
 - 3.2. Запрос на удаление группы записей.
 - Удалить все записи из таблицы *Vacav*, относящиеся к предприятию с кодом 105.
 - 3.3. Запрос на создание таблицы
 - Создать новую таблицу *Work1*, структура которой должна совпадать со структурой таблицы *Work*. Добавить в нее все записи об инженерах. Используя запрос на обновление, изменить значение поля *Код профессии* во всех записях на значение 14.
 - 3.4. Запрос на добавление записей
 - Добавить в таблицу *Work* все записи из таблицы *Work1*.
4. Перекрестные запросы
- Определить количество вакансий по каждой профессии на каждом предприятии. Результирующая таблица должна иметь вид, изображенный на рис. 3.

Предприятия	Бухгалтер	Инженер	Менеджер	...
Автобаза	1	2	1	
Банк		1	2	
Магазин	1		3	
...				

Рис. 3. Перекрестная таблица «Вакансии службы занятости»

- Вывести перекрестный список работников и профессий, по которым работники ищут работу, в виде таблицы, изображенной на рис. 4.

Работник	Количество профессий	Бухгалтер	Инженер	Менеджер	...
Андреев О.	2	1	1		
Васильев П.	1				
Иванов К.	2			1	
...					

Рис. 4. Перекрестная таблица «Предложения службы занятости»

Табличный язык запросов QBE используется для создания следующих видов запросов: запросов выбора, параметрических запросов, запросов действия и перекрестных запросов.

- *Запросы выбора* извлекают из таблицы данные, удовлетворяющие заданным в запросе критериям отбора, и отображают их в выходном динамическом наборе.
- *Параметрические запросы* содержат параметрические критерии отбора записей. Значения этих критериев задает пользователь во время выполнения запроса.
- *Запросы действия* изменяют информацию в таблицах базы данных: удаляют группу записей, удовлетворяющих заданному критерию отбора, обновляют записи или добавляют записи в другую таблицу.
- *Перекрестные запросы* объединяют числовые или денежные данные в перекрестную таблицу, у которой в качестве имен полей и строк используются значения исходной таблицы или запроса, а значения ячеек формируются с помощью функций.

Запросы в Access можно создать с помощью мастера или вручную в режиме конструктора. Для создания запроса в режиме конструктора надо добавить в окно запроса таблицы, участвующие в запросе. Затем заполнить запросную таблицу, указав в ней все поля, участвующие в запросе, условия отбора записей, порядок и способ сортировки. При необходимости указать групповую операцию (меню *Вид, Группировка*). Для вычисления итоговых значений используются встроенные статистические функции Sum, Count, Max, Min, Avg. Для переименования заголовков столбцов в выходном наборе перед именем поля в бланке запроса вводится новое имя, отделенное от имени поля двоеточием. Для вывода в выходной набор вычисляемого значения в бланке запроса надо создать новое поле. В ячейку *Поле этого* пустого столбца ввести выражение, перед которым записать имя нового поля, например,

Сумма: [Количество вакансий]*[Зарплата]

При разработке параметрического запроса выбора введите в ячейку строки *Условие отбора* параметризуемого поля текст обращения к пользователю, заключив текст в квадратные скобки.

При создании перекрестных запросов используйте мастера перекрестных запросов. Если в перекрестной таблице должны использоваться поля из нескольких таблиц, предварительно создайте запрос выбора, включив в него необходимые поля. На основе созданного запроса выбора создайте перекрестный запрос. При необходимости перекрестный запрос можно доработать вручную в режиме конструктора.

Для создания запроса действия выполните следующие операции:

- Создайте запрос выбора, включив в него все поля, необходимые для запроса действия.
- Выполните созданный запрос, чтобы убедиться в правильном отборе записей.
- Преобразуйте запрос выбора в запрос действия, выбрав команду *Добавить, Удалить, Обновить* или *Создание таблицы* в меню *Запрос*.

ЛАБОРАТОРНАЯ РАБОТА N5

SQL ЗАПРОСЫ

1. Запросы выбора

- Создать и выполнить QBE запрос, выводивший коды всех муниципальных округов, в которых проживают зарегистрированные в службе занятости работники.
- Изменить SQL текст созданного запроса таким образом, чтобы исключить из выходного набора повторяющиеся коды округов.

2. Запросы объединения

- 2.1. Запрос объединения на основе двух таблиц
Вывести список всех кодов муниципальных округов, в которых имеются клиенты службы занятости (работники или предприятия).
- 2.2. Запрос объединения на основе двух запросов
Вывести общий список наименований профессий, в который должны входить профессии зарегистрированных работников и профессии, необходимые предприятиям.

3. Запросы с подчиненным запросом

- 3.1. Использование подчиненного запроса в качестве критерия отбора записей

Найти все предприятия, в которых предлагаемая программистам зарплата выше средней зарплаты, предлагаемой службой занятости. В выходной набор вывести наименования предприятий и предлагаемую ими зарплату. Преобразовать разработанный запрос в запрос с параметром «Наименование профессии».

- 3.2. Использование параметров ALL и ANY в подчиненных запросах
- Вывести информацию о предлагаемых профессиях, для которых зарплата больше максимальной зарплаты программиста (Код профессии=11). Выходная таблица должна иметь вид, изображенный на рис. 5.
 - Вывести информацию о предлагаемых профессиях, для которых зарплата больше минимальной зарплаты программиста.

Профессия	Предприятие	Зарплата
Бухгалтер	Магазин	10000,00 р.
...		

Рис. 5. Выходной набор запроса с подчиненным запросом

- Вывести информацию о предлагаемых профессиях, для которых зарплата больше минимальной зарплаты программиста.

3.3. Использование подчиненного многотабличного запроса

Вывести информацию о предлагаемых профессиях, для которых зарплата больше максимальной зарплаты программиста. Используйте в подчиненном запросе наименование профессии.

4. Управляющие запросы

- Написать SQL запрос для создания таблицы Kurs, содержащей информацию о курсах переподготовки кадров. Структура таблицы Kurs должна иметь вид, изображенный на рис. 6.

Код курса	Код профессии	Продолжительность	Дата начала

Рис. 6. Структура таблицы Kurs

- Установить первичный ключ таблицы Kurs, используя команду ALTER для создания первичного индекса.
- Установить связь таблицы Kurs с таблицей Prof, используя команду ALTER для создания вторичного индекса.
- Добавить в таблицу Kurs запись
1, 14, 2, #25/01/1999#
используя команду INSERT INTO.
- Создать таблицу List, содержащую информацию о работниках, посещающих курсы переподготовки кадров. Структура таблицы должна иметь вид, изображенный на рис. 7.

Номер работника	Код курса	Дата начала

Рис. 7. Структура таблицы List

- Установить первичный ключ таблицы List
- Установить связи таблицы List с другими таблицами базы данных.
- Добавить в таблицу List новое поле *Аттестация* с логическим типом значения.

- Создать QBE запрос для нахождения всех работников, зарегистрированных в службе занятости и имеющих профессию инженера. В выходной набор вывести номера работников и два вычисляемых поля: Курс со значением 1 и Дата начала со значением 25.01.1999. В выходной набор поместить только 5% всех отобранных записей, установив значение свойства запроса *Набор значений* в 5%. Сохранить запрос под именем «Курсы».
- Создать SQL запрос на добавление в таблицу List записей выходного набора запроса «Курсы».

Методические указания

В СУБД Access используется диалект языка SQL – SQL Microsoft Jet. Язык SQL позволяет создавать все виды запросов, которые поддерживаются языком QBE, а также запросы, которые нельзя создать с помощью языка QBE: запросы объединения, запросы с подчиненным запросом, управляющие запросы, запросы к серверу.

Запрос объединения объединяет значения полей из двух или нескольких таблиц в один список. Синтаксис запроса объединения:

<Запрос 1>UNION<Запрос 2>

Запрос с подчиненным запросом позволяет задавать сложные условия отбора записей или сложное выражение для вычисляемого поля.

Управляющие запросы изменяют структуру таблицы, создают или удаляют таблицу и индексы таблицы.

Запросы к серверу позволяют работать с базами данных, размещенными на сервере.

В Access все QBE запросы преобразуются в SQL запросы, тексты которых можно увидеть и отредактировать в режиме запроса SQL (меню Вид, SQL). Синтаксис основных команд SQL приведен в Приложении 2.

Для создания запроса SQL выполните следующие действия

- Выберите в окне базы данных закладку *Запрос*.
- Нажмите кнопку *Создать*.
- Выберите режим *Конструктор*.
- Не добавляя таблицы или запроса, закройте окно *Добавление таблиц*.
- Выберите команду *Запрос SQL* в меню *Запрос* и одну из подкоманд *Объединение*, *Управление* или *Запрос к серверу*.
- Введите команду SQL.

При использовании подчиненного SQL запроса в качестве критерия отбора записей в QBE запросе введите команду SQL в ячейку *Условие отбора* соответствующего столбца. Синтаксис подчиненного запроса:

<Знак операции сравнения> [ANY|ALL] (Команда SELECT)

Параметр ANY используется для отбора записей в главном запросе, которые удовлетворяют сравнению хотя бы с одной записью, отобранной в подчиненном запросе. Например, команда SQL для поиска всех товаров, цена которых больше, чем цена хотя бы одного товара, проданного со скидкой 25%, имеет вид:

```
SELECT * FROM Товары
WHERE Цена > ANY (SELECT Цена FROM Заказано
WHERE Скидка >= 0.25);
```

Параметр ALL используется для отбора только тех записей в главном запросе, которые удовлетворяют сравнению со всеми записями, отобранными в подчиненном запросе. Например, команда SQL для поиска всех товаров, цена которых больше, чем цена всех товаров, проданных со скидкой 25%, имеет вид:

```
SELECT * FROM Товары
WHERE Цена > ALL (SELECT Цена FROM Заказано
WHERE Скидка >= 0.25);
```

Для удобства ввода длинной команды SQL можно открыть окно *Область ввода*, нажав клавиши Shift/F2. При использовании подчиненного запроса для создания вычисляемого поля команда SQL вводится в ячейку *Поле*. Для создания сложного подчиненного SQL-запроса на основе нескольких таблиц разработайте QBE-запрос, выполняющий ту же функцию. SQL-текст разработанного запроса можно использовать как команду SELECT подчиненного запроса. Для этого скопируйте и вставьте его в текст разрабатываемого запроса с подчиненным запросом, а затем отредактируйте запрос.

ЛАБОРАТОРНАЯ РАБОТА №6

СОЗДАНИЕ ЭКРАННЫХ ФОРМ

1. Однотабличные формы

1.1. Создание автоформы

Создать колончатую форму для изменения и ввода данных о муниципальных округах на основе таблицы *Округ*.

Испытайте форму. Добавьте с помощью формы новые записи в таблицу *Округ*.

1.2. Изменение свойств формы

Изменить свойства формы, созданной в пункте 1.1.

- Озаглавьте форму «Список округов».
- Удалите линии границ между разделами формы.
- Измените цвет фона формы.

- Испытайте форму.
- Установите свойство *Ввод данных* таким образом, чтобы форма могла быть использована только для ввода данных.
- Испытайте форму.

1.3. Добавление в форму поля со списком

Создать форму для изменения и ввода информации о работнике. Вид формы приведен на рис. 8.

Номер работника	
Фамилия	Имя
Имя	Карп
Отчество	Петрович
Дата рождения	11.12.45
Адрес	117480, Песочная 32-105
Телефон	422.69.11
Округ	Таганский
Дата регистрации	11.02.99

Рис. 8. Однотабличная форма «Карта»

- Создайте форму на основе таблицы *Рис*, разместив в ней все поля таблицы.
 - Озаглавьте форму «Информация о работнике».
 - Замените в форме элемент *Поле* «Код Мо» на элемент управления *Поле со списком* для выбора кода муниципального округа по его наименованию. Для создания элемента управления *Поле со списком* используйте мастера, выбрав его на панели инструментов. Скройте поле *Код Мо* в форме, задав нулевую ширину столбца *Код Мо* в строке списка.
 - Измените при необходимости порядок обхода элементов формы, выполнив команду *Последовательность переходов* меню *Вид*.
 - Сохраните форму под именем «Карта».
- #### 2. Многотабличные формы
- ##### 2.1. Форма на основе многотабличного запроса
- Создать форму для просмотра вакансий всех предприятий. Вид формы представлен на рис. 9.
- Создайте многотабличный запрос и сохраните его.
 - Создайте форму на основе запроса.

- Измените свойства формы таким образом, чтобы она могла быть использована только для просмотра данных.

Рис. 9. Многотабличная форма на основе запроса

2.2. Форма на основе нескольких таблиц

2.2.1. Обычная многотабличная форма

Создать форму для просмотра предложений работников. Вид формы приведен на рис. 10.

Рис. 10. Обычная многотабличная форма

- Создайте форму с помощью мастера.
 - Укажите мастеру все поля, которые должны быть размещены в форме.
 - Измените в режиме конструктора свойства формы и ее элементов.
- ### 2.2.2. Многотабличная форма с раскрывающейся подчиненной формой

Создать многотабличную форму с раскрывающейся подчиненной формой для просмотра и изменения вакансий предприятий. Разместить в главной форме все поля таблицы Company, а в подчиненной форме - поля таблицы Vacanp. Записи подчиненной формы должны быть связаны с текущей записью главной формы. Вид главной формы приведен на рис. 11.

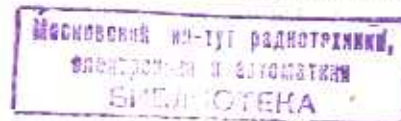
Рис. 11. Главная форма

Подчиненная форма должна открываться при нажатии на кнопку *Вакансии* и иметь вид, изображенный на рис. 12.

Профессия	Опыт работы	Зарплата	Вакансии
Инженер	<input checked="" type="checkbox"/>	279,00р	3
Менеджер	<input checked="" type="checkbox"/>	44,00р	3
программист	<input type="checkbox"/>	2 000,00р	3
Слесарь	<input type="checkbox"/>	59,00р	5

Рис. 12. Подчиненная форма

- Используйте мастера для построения связанных форм.
- Укажите мастеру все поля, которые должны быть размещены в двух формах, указав поле *Код работника* только один раз.
- Укажите мастеру главную таблицу (по Company).
- Установите переключатель в значение *Связанные формы*.
- Испытайте форму.
- Для удобства ввода кода профессии замените в подчиненной форме управляющий элемент *Поле* «Код профессии» на элемент *Паге со*



списком.

Сохраните форму под именем «Вакансии предприятий».

2.2.3. Многотабличная форма с вложенной подчиненной формой

Создать многотабличную форму для просмотра предложений работников. Вид формы приведен на рис. 13.

Рис. 13. Многотабличная форма с вложенной формой

- Создайте главную форму на основе таблицы *F10*.
- Создайте подчиненную табличную форму без заголовка на основе таблиц *Work* и *Prof*.
- Выберите главную форму и разместите в ней элемент управления *Подчиненная форма*.
- Установите для элемента *Подчиненная форма* следующие свойства: *Объект-источник* (имя подчиненной формы), *Подчиненные поля* (имена полей связи подчиненной формы), *Основные поля* (имена полей связи главной формы).
- Испытайте форму.
- Измените свойства формы и элементов формы для придания ей указанного на рис. 13 вида. Установите свойства формы таким образом, чтобы она могла быть использована только для просмотра информации.
- Сохраните форму под именем «Предложения работников».

3. Управляющая форма

Разработать управляющую форму с двумя кнопками для выполнения запросов с параметрами «Вакансии!» и «Работники!», созданных в лабораторной работе 4. Вид формы приведен на рис. 14.

- Создайте в режиме конструктора пустую форму.
- Установите в свойстве *Подпись* значение «Поиск».

Рис. 14. Управляющая форма

- Разместите в форме кнопку.
 - Работая с мастером, создайте оформление кнопки и укажите действие *Запуск запроса*, которое должно быть выполнено при возникновении события *Нажатие кнопки*.
 - Разместите вторую кнопку.
 - Измените оформление формы.
- #### 4. Использование макросов при разработке форм
- ##### 4.1. Использование макроса для открытия формы
- Добавить в форму «Предложения работников» кнопку, нажатие которой будет открывать форму «Карта» с дополнительной информацией о работнике.
- Создайте макрос с макрокомандой *Открыть форму*. Синхронизируйте форму «Карта» с формой «Предложения работников» таким образом, чтобы в форме «Карта» отображалась запись о работнике, выбранном в форме «Предложения работников».
 - Разместите кнопку в форме «Предложения работников».
 - Укажите в свойстве кнопки *Нажатие кнопки* имя разработанного макроса.
- ##### 4.2. Использование макросов для изменения и отображения в форме полей таблицы
- Разработать форму для вывода полей таблицы *Vacant* и обновления полей *Зарплата*, *Количество вакансий* и *Сумма*. Вид формы изображен на рис. 15. Значение поля *Сумма* должно вычисляться согласно выражению: $Зарплата * [Количество\ вакансий]$. Поле *Сумма* должно автоматически обновляться в таблице *Vacant* при возникновении одного из событий: *текущая запись*, *обновление поля Зарплата*, *обновление поля Количество вакансий*.



Рис. 15. Форма для обновления полей

- Добавьте в таблицу *Вакасии* поле *Сумма*.
- Создайте форму «Вакасии». Элементы управления *Код предприятия*, *Код профессии* и *Сумма* сделайте недоступными для изменения пользователем.
- Установите свойства формы таким образом, чтобы запретить ее использование для добавления новых записей.
- Разработайте макрос для задания значения полю *Сумма*. Используйте в макросе макрокоманду *Задать значения*.
- Свяжите разработанный макрос со свойством *После обновления полей* *Зарплата* и *Количество вакансий* и со свойством формы *Текущая запись*.

Методические указания

Экранные формы используются для просмотра, изменения и ввода данных. Формы могут быть созданы на основе таблицы, запроса или на основе нескольких источников данных.

Формы можно разрабатывать с помощью мастера форм, вручную в режиме конструктора или комбинируя эти два способа. При разработке формы вручную в ней необходимо разместить элементы управления – объекты формы, которые служат для вывода данных, выполнения макрокоманд или оформления. Чтобы добавить элемент управления в форму, его надо выбрать на панели инструментов и перетащить в форму.

Все элементы управления, как и сама форма, обладают свойствами. Свойства определяют внешний вид и поведение формы (ее элементов). Для установки или изменения свойства управляющего элемента надо выбрать элемент и выполнить команду *Свойства* из контекстного меню. Свойства формы будут доступны, если два раза щелкнуть по области определения формы – квадрату на пересечении линеек в режиме конструктора.

Для создания в форме *Поля со списком* для ввода значения из заранее определенного списка (например, кода муниципального округа для записи таблицы *Гео* из списка муниципальных округов таблицы *Округ*) надо выполнить следующие действия.

- Включите мастера на панели инструментов.
- Выберите элемент *Поле со списком* на панели инструментов и перетащите его в форму.
- Действуйте по указанию мастера.
- Укажите мастеру на необходимость скрыть столбец с первичным ключом.
- Для ввода выбранного в списке значения в поле базовой таблицы, для которого и создается поле со списком, укажите мастеру на необходимость сохранения выбранного значения в поле базовой таблицы.

Многотабличные формы, содержащие поля нескольких таблиц, могут быть созданы на основе предварительно разработанного запроса или на основе нескольких таблиц. Обычная многотабличная форма содержит поля из нескольких таблиц, не связанных отношением «Главная – подчиненная». Такую форму удобно создавать с помощью мастера.

Многотабличная форма со связанными формами содержит главную форму, основанную на главной таблице, и подчиненные формы. В главной форме отражается одна запись главной таблицы, а в подчиненных – данные других таблиц, связанные с записью главной формы. Зависимая форма может быть встроена в главную форму или раскрываться при возникновении в главной форме некоторого события, например, события *Нажатие на кнопку*. При разработке подчиненной формы не следует размещать в ней поля связи, если они размещены в главной форме.

Макрос в Access представляет собой последовательность макрокоманд, задающих последовательность операций с базой данных. Макрос создается в специальном окне макросов, которое можно открыть, выбрав закладку *Макросы* в окне *Базы данных*. Для каждой макрокоманды макроса, которую можно выбрать из списка, указываются ее аргументы. Количество и тип аргументов зависят от используемой макрокоманды. Для некоторых макрокоманд аргументы можно выбрать из списка. Для задания в макрокоманде условия отбора записей из таблицы, на основе которой создана текущая форма, используется следующий синтаксис:

Поле текущей формы = *Элемент управления другой формы*, где *Элемент управления другой формы* определяет значение для отбора записей из текущей таблицы и задается в следующем виде:

Forms! [*Имя другой формы*]! [*Имя элемента управления*]

ЛАБОРАТОРНАЯ РАБОТА N7

СОЗДАНИЕ ОТЧЕТОВ

1. Простые отчеты

Создать колончатый отчет для печати выбранной в форме «Карта» записи таблицы F10.

- Используйте режим автоотчета.
- Озаглавьте отчет «Карточка работника».
- Сохраните отчет под именем «Отчет1».
- Создайте макрос, который открывает отчет с данными о работнике, выбранном в форме «Карта». В микрокоманде макроса укажите условие отбора записи из таблицы F10, со значением поля *Номер работника*, совпадающим со значением элемента управления *Поле Номер работника* формы. Синтаксис выражения для отбора записей имеет вид `[Имя поля таблицы]=Tochka![Имя формы]![Имя элемента управления]`

Укажите режим *Просмотр* для микрокоманды.

- Сохраните макрос.
- Добавьте в форму «Карта» кнопку с надписью «Печать карточки». В свойстве кнопки *Нажатие кнопки* укажите имя разработанного макроса.
- Испытайте форму.
- Замените в макросе режим *Просмотр* на *Печать*.
- Распечатайте карточку работника.

2. Отчет с группированием

Создать табличный отчет «Регистрация работников» на основе таблицы F10 с группированием данных по полю *Дата регистрации*. Вид отчета приведен на рис. 16.

- Укажите интервал группировки *По месяцам*.
- Укажите порядок сортировки: внутри даты регистрации по фамилии, имени, отчеству.
- В примечании отчета укажите общее количество зарегистрированных работников.
- Создайте сквозную нумерацию записей отчета.
- Сохраните отчет под именем «Регистрация работников».

3. Многотабличный отчет

Создать табличный отчет «Вакансии предприятий». Вид отчета приведен на рис. 17.

- Включите в отчет информацию обо всех зарегистрированных в службе занятости предприятиях.

Дата регистрации	Фамилия	Имя	Отч.	Дата регистрации	Дата рождения	Текст
Февраль 1999						
1	Васильев	Петр	Иванович	11.02.99	9 марта 1970 г.	887
2	Иванов	Евг.	Петрович	11.02.99	14 декабря 1945 г.	421
3	Петров	Петр	Сергеевич	11.02.99	5 февраля 1955 г.	234
Март 1999						
4	Александров	Олег	Сергеевич	15.03.99	1 января 1976 г.	123
5	Большаков	Ольга	Николаевна	17.03.99	12 января 1957 г.	120
Общее количество			5			

Рис. 16. Однотабличный отчет с группированием

Месяц	Тип трудовой	Профессия	Зарплата	Вакансии
Май	Летоблаги	бухгалтер	10 000,00р	4
		степлер	700 000,00р	1
		Итого:		
Май	Хлебосыбытки	Итого:	779,00р	3

Рис. 17. Многотабличный запрос с группированием

- В качестве источника данных для отчета используйте многотаб-

личный запрос.

- Сгруппируйте данные по наименованию муниципальных округов, а внутри округа по наименованию предприятий.
- При разработке отчета используйте мастера.
- Доработайте отчет вручную.
- Укажите для каждого округа суммарное количество вакансий, добавив в отчет раздел *Примечание группы* и разместив в нем вычисляемое поле.
- Разместите в примечании отчета итоговые поля: *Общее количество вакансий, Средняя зарплата, Максимальная зарплата, Минимальная зарплата*.
- Сохраните отчет под именем «Вакансии предприятий».

4. Отчет в виде стандартного письма

Разработать макет стандартного письма для уведомления работников о найденных для них вакансиях.

Московская служба занятости

Дата: *Текущая дата*

Адресат: *Фамилия И.О.*

Содержание: Информация о вакансиях.

Уважаемый *Фамилия Имя Отчество!*

Мы рады предоставить Вам информацию об интересующих Вас вакансиях. Вы можете ознакомиться с информацией. *Начальная дата* по *Конечная дата*.

С уважением,

Васильев В.В.

генеральный директор

- В качестве источника данных для отчета используйте многотабличный запрос, выводящий поля *Фамилия, Имя, Отчество* для работников, профессии которых требуются на предприятиях. При разработке запроса укажите дополнительную связь между таблицами *Work* и *Vacant* по полям *Код профессии*. Установите свойство запроса *Уникальные значения* в «Да», чтобы исключить многократный вывод в выходной набор работников, для которых нашлось несколько вакансий.
- Разрабатывайте отчет вручную. Удалите в макете отчета все лишние разделы. Элементы отчета *Поля* и *Подписи* разместите в области данных.
- В полях *Начальная дата* и *Конечная дата* укажите выражения $=Date()+2$ и $=Date()+7$.

- В полях *Фамилия И.О.* и *Фамилия Имя Отчество* укажите текстовые выражения.
- Выберите тип шрифта и его размер.
- Для печати каждого письма на отдельной странице установите в конце письма элемент управления *Разрыв страницы*.
- Сохраните отчет под именем «Письмо».
- Разработайте макрос для открытия и печати отчета «Письмо». Сохраните макрос под именем «Письмо». Запустите макрос.

Методические указания

Отчет позволяет организовать данные в определенном порядке и вывести их на печать. Стандартная структура отчета приведена на рис. 18.

Заголовок отчета	Появляется в начале отчета. Содержит заголовок отчета и текущую дату.
Верхний колонтитул	Располагается в начале каждой страницы. Используется для заголовков столбцов.
Заголовок группы	Располагается перед первой записью каждой группы. Используется для заголовка группы.
Область данных	Появляется для каждой записи отчета.
Примечание группы	Располагается после последней записи группы. Используется для итоговых полей группы.
Нижний колонтитул	Располагается в конце каждой страницы. Используется для размещения номера страницы.
Примечание отчета	Появляется в конце отчета. Используется для итоговых данных отчета.

Рис. 18. Стандартная структура отчета

При разработке макета отчета некоторые разделы отчета можно удалить или добавить, используя меню *Вид*.

Отчеты можно создавать автоматически (автоотчеты), вручную в режиме конструктора и с помощью мастера отчетов. Источником данных для отчета может быть таблица, запрос, а также несколько таблиц или запросов.

Для создания итоговых полей в отчете используется элемент управления *Поле*, в свойстве которого *Данные* указывается итоговое выражение, например, $=Count([Номер работника])$.

Для сквозной нумерации записей отчета используется элемент управления *Поле*. Этот элемент надо разместить в области *Данные* и

указать в его свойстве *Данные* выражение =1, а в свойстве *Сумма с накоплением* указать значение «Для всего».

ЛАБОРАТОРНАЯ РАБОТА №8

РАЗРАБОТКА ПРИЛОЖЕНИЯ

Разработать приложение «Служба занятости». Приложение должно осуществлять управление службой занятости и предоставлять пользователям следующие возможности по обработке данных:

- регистрация работников, ищущих работу;
 - регистрация вакансий предприятий;
 - просмотр и редактирование данных о работниках и предприятиях;
 - выполнение запросов о поиске вакансий для работников и работников для предприятий;
 - получение отчетов о зарегистрированных в службе занятости работниках и вакансиях предприятий;
 - изменение и редактирование словарей по профессиям и муниципальным округам.
1. Использование макросов для создания меню

Создать главное меню приложения, вид которого приведен на рис. 19.

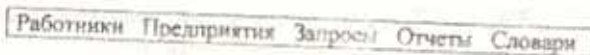


Рис. 19. Главное меню приложения

Пункт меню *Работники* должен содержать команды *Просмотр* и *Редактирование*, пункт *Предприятия* – команды *Просмотр* и *Редактирование*, пункт *Запросы* – команды *Поиск вакансий* и *Поиск работников*, пункт *Отчеты* – команды *Работники* и *Вакансии*, пункт *Словари* – команды *Профессии* и *Округа*.

- Создайте макросы для выполнения команд пункта *Работники*, указав в макросе для команды *Просмотр* макрокоманду *ОткрытьФорму* с аргументом *Имя формы* «Предложения работников», а в макросе для команды *Редактирование* – макрокоманду *ОткрытьФорму* с аргументом *Имя формы* «Карта». Сохраните макросы под именами «Просмотр_р» и «Редакт_р».
- Создайте макрогруппу, объединяющую макросы команды пункта меню *Работники*. Для создания макрогруппы выполните следующие действия:
- выберите закладку *Макросы*, нажмите кнопку *Создать*;

- активизируйте режим ввода имен макроса, выбрав пиктограмму *Имена макросов (XYZ)*;
 - укажите в ячейках колонки *Имя макроса* наименования команд пункта меню (*Просмотр* и *Редактирование*), в ячейках колонки *Макрокоманда* макрокоманды для каждого макроса (*ЗапускМакроса*), а в каждой ячейке колонки *Примечание* текст для строки статуса;
 - внесите для каждой макрокоманды имя макроса в аргументы макрокоманды («Просмотр_р» и «Редакт_р»);
 - сохраните макрогруппу под именем «Работники».
 - Составьте макрос, формирующий и обслуживающий пользовательскую строку меню, указав в нем макрокоманду *ДобавитьМеню* с аргументами *Название меню* («Работники»), *Имя макроса* («Работники»).
 - Сохраните макрос под именем «Меню».
 - Создайте макрогруппы для каждого пункта меню и добавьте в макрос «Меню» макрокоманды *ДобавитьМеню* для каждого пункта меню.
2. Подключение меню к форме

Разработайте новую пустую форму. Для подключения разработанного меню к форме установите в свойстве формы *Строка меню* имя макроса для пользовательской строки меню («Меню»). Установите другие свойства формы и ее разделов таким образом, чтобы скрыть в ней полосы прокрутки, номера записей, область выделения, разделительные линии, кнопки развертывания и свертывания окна. Измените цвет области данных. Сохраните форму под именем «Служба занятости». Вид формы приведен на рис. 19. Испытайте форму.

ПРИЛОЖЕНИЕ 1

ВАРИАНТЫ ОПИСАНИЯ ПРЕДМЕТНОЙ ОБЛАСТИ

1. Служба занятости

Служба занятости предоставляет информацию о предложениях работников и вакансиях предприятий.

Информация о предложении работника: регистрационный номер работника, фамилия, имя, отчество, дата рождения, адрес, муниципальный округ (код муниципального округа, наименование муниципального округа, административный округ), список профессий, которыми владеет работник (код профессии, наименование профессии, наличие опыта работы по профессии).

Информация о вакансии предприятия: код предприятия, наименование предприятия, адрес, муниципальный округ (код муниципального округа, наименование муниципального округа, административный округ),

список профессий, по которым имеются вакансии на предприятии (код профессии, наименование профессии, наличие опыта работы по профессии, количество вакансий по профессии).

2. Магазины

Магазин имеет отделы, в которых продаются товары. Информация о каждом отделе содержит номер отдела, список работников, список продаваемых в отделе товаров. Информация о товаре: код товара, наименование товара, цена магазина, отпускная цена изготовителя, информация об изготовителе. Информация об изготовителе: наименование фирмы, адрес фирмы, поставляемые товары, отпускные цены. Информация о работнике: табельный номер, фамилия, имя, отчество, адрес, телефон, номер отдела. Работник может работать только в одном отделе. Один и тот же товар может продаваться в нескольких отделах.

3. Индивидуальные учебные планы студентов

Часть студентов факультета обучается по индивидуальным планам. Каждый студент имеет фамилию, имя, отчество, специальность и уникальный учетный номер. Каждый студент может выбирать для изучения в данном семестре дисциплины из списка. Список дисциплины составляется для каждой специальности по каждому семестру. Для каждой дисциплины в плане ее изучения указываются наименование дисциплины, код дисциплины, специальность, номер семестра, а для каждого семестра — количество часов в неделю лекций, практических занятий, лабораторных работ и вид отчетности в конце семестра. Для каждой специальности указываются ее код и наименование.

4. Библиотека

Читатели библиотеки получают на время книги в библиотеке. Каждая операция выдачи/дачи читателем книги должна быть зафиксирована: указан читатель, книга, вид операции, дата выполнения операции. Каждый читатель имеет номер читательского билета, фамилию, имя, отчество, адрес, телефон. Каждая книга имеет авторов, наименование, год издания, шифр. Каждый экземпляр книги имеет учетный номер.

5. Торговая фирма

Филиалы торговой фирмы, имеющие наименование, адрес и телефон, продают различные виды товаров. Для каждого филиала ведется учет текущего количества каждого вида товаров. Товар в филиалы поставляют поставщики. Каждый поставщик может поставлять несколько видов товара. Разные поставщики могут поставлять одинаковые виды товаров по разным ценам. Филиал может получать одинаковый товар у нескольких поставщиков. Информация о поставщике содержит наименование поставщика, адрес, телефон, список поставляемых видов товаров с указанием их цены.

6. Трудоустройство выпускников вузов

В службе занятости о каждом вузе имеется следующая информация: наименование вуза, адрес, телефон, список выпускников. Для каждого выпускника в списке указаны фамилия, имя, отчество, специальность, средний балл. Для каждой фирмы, заказывающей специалистов, имеется следующая информация: наименование фирмы, адрес, телефон, список вакансий. Информация о каждой вакансии: специальность, зарплата, количество вакансий. Специальность имеет код и наименование.

7. Расписание экзаменов

Каждый студент вуза сдает экзамены по ряду дисциплин согласно расписанию экзаменов. В расписании экзаменов для каждой студенческой группы указана следующая информация: дата, время, аудитория, преподаватель. Расписание составлено таким образом, что студенческая группа может сдавать в один день только один экзамен, а в одной аудитории не может проходить несколько экзаменов одновременно. Каждая группа имеет шифр и список студентов. Каждый студент имеет шифр, фамилию, имя и отчество.

8. Картоотека преподавателей

В картоотеке преподавателей содержится информация о преподавателях института. Для каждого преподавателя указываются учетный номер, фамилия, имя, отчество, факультет, кафедра, список преподаваемых им дисциплин. Для каждой дисциплины из списка указываются наименование дисциплины, ее код, специальность, факультет, год преподавания. Преподаватели института могут работать только на одной кафедре. Кафедра принадлежит одному факультету. Преподаватель может преподавать дисциплины на нескольких факультетах. Факультет имеет код, наименование и список принадлежащих ему кафедр. Кафедра имеет код и наименование.

9. Оптовая фирма

Фирма закупает партии товаров у поставщика и размещает их в хранилищах. Информация о каждой партии товара: вид товара, поставщик, закупочная цена, количество, дата закупки. В каждом хранилище, имеющем уникальный номер, могут храниться несколько партий товара, но только одного вида. Товар одного вида может храниться в нескольких хранилищах, но товары одной партии могут храниться только в одном хранилище. Поставщик имеет наименование, адрес, телефон и может поставлять несколько видов товара.

10. Результаты экзаменационной сессии

Студенты каждый семестр сдают экзамены по нескольким дисциплинам. Студент может пересдавать экзамен несколько раз: разным преподавателям. В один день студент не может сдавать несколько экзаменов. Информация о всех экзаменах каждого студента должна

храниться в базе данных. Информация о студенте: шифр, фамилия, имя, отчество, группа. Информация о каждом экзамене: наименование дисциплины, код дисциплины, семестр, дата сдачи, преподаватель.

11. Поставка деталей

Завод получает комплектующие детали от разных поставщиков. В базе данных хранится информация о поставляемых деталях: вид детали, поставщик, количество деталей от каждого поставщика с указанием даты поставки. Каждый поставщик может поставлять несколько видов деталей. Поставщики могут поставлять одинаковые виды деталей. Информация о каждом поставщике: наименование поставщика, адрес, телефон, список поставляемых видов деталей. Элемент списка содержит код вида, наименование.

12. Учебные планы института

Учебные планы составляются для каждой учебной группы. В планах указываются факультет, специальность, группа, семестр, дисциплина, вид отчетности в конце семестра (зачет, экзамен, зачет по курсовой работе), количества часов лекций, лабораторных работ, практических и индивидуальных занятий в неделю. Факультет может обучать студентов по нескольким специальностям. По одной специальности могут обучаться несколько групп, а по некоторым специальностям могут готовить специалистов несколько факультетов. Некоторые дисциплины изучаются несколько семестров. Специальность имеет код и наименование. Факультет имеет код и наименование.

ПРИЛОЖЕНИЕ 2

ОСНОВНЫЕ КОМАНДЫ SQL

Команды обработки данных

1. Команда SELECT

```
SELECT [ALL|DISTINCT|DISTINCTROW] список полей для вывода
FROM список таблиц, из которых отбираются данные
[WHERE условие отбора]
[GROUP BY список полей, по которым группируются записи]
[HAVING критерии поиска]
[ORDER BY список полей, задающих порядок сортировки].
```

Пример однотабличного запроса

```
SELECT Фамилия, Имя, [Дата рождения]
FROM FIO
WHERE ([Код Мо]=103) AND ([Дата рождения]>=#01/01/1960#)
ORDER BY Фамилия, Имя.
```

Пример многотабличного запроса

```
SELECT Фамилия, Имя, [Наименование Мо] AS MO, Ao
FROM FIO INNER JOIN OKRUG
ON FIO.[Код Мо]=OKRUG [Код Мо].
```

Пример запроса объединения двух запросов

```
SELECT FIO.[Код Мо]
FROM FIO
UNION
SELECT COMPANY [Код Мо]
FROM COMPANY;
```

2. Команда добавления записи в таблицу

```
INSERT INTO имя таблицы [(список полей)]
VALUES (список значений);
```

Пример

```
INSERT INTO FIO ([Номер работника], Фамилия, Имя, [Дата рождения])
VALUES (25, 'Ермолаев', 'Максим', #05/09/1970#).
```

3. Команда добавления в таблицу группы записей из другой таблицы или запроса

```
INSERT INTO имя таблицы-получателя (список полей)
SELECT список полей источника записей
FROM имя таблицы или запроса - источника записей;
```

4. Команда изменения поля записи

```
UPDATE имя таблицы
SET имя поля=выражение
WHERE условие отбора.
```

Пример

```
UPDATE VACAN
SET Зарплата=Зарплата*1.5
WHERE [Код профессии]=11.
```

5. Команда удаления записей

```
DELETE FROM имя таблицы
WHERE условие отбора.
```

Команды определения данных

1. Команда создания таблицы

```
CREATE TABLE имя таблицы (поле1 тип поля1, ..., полеN тип поляN);
```

Пример

```
CREATE TABLE OKRUG ([Код Мо] SHORT, [Наименование Мо]
TEXT(20), Ao TEXT(3));
```

2. Команда создания первичного индекса

```
ALTER TABLE имя таблицы
ADD CONSTRAINT имя первичного индекса
```

PRIMARY KEY (список полей первичного индекса).

3. Команда создания вторичного индекса и установления связи с внешней таблицей

ALTER TABLE имя таблицы

ADD CONSTRAINT имя вторичного индекса

FOREIGN KEY (список полей внешнего ключа)

REFERENCES имя внешней таблицы (список связанных полей внешней табл. цы);

Пример

ALTER TABLE FIO

ADD CONSTRAINT индекс2

FOREIGN KEY ([Код Мо])

REFERENCES OKRUG ([Код Мо]);

4. Команда добавления поля

ALTER TABLE имя таблицы

ADD COLUMN поле тип поля;

5. Команда удаления поля

ALTER TABLE имя таблицы

DROP COLUMN поле;

Библиографический список

1. Ульман Дж. Основы систем баз данных - М.: Финансы и статистика, 1983. - 334с.
2. Access 7.0 для Windows 95. - К.: Торгово-издательское бюро ВНУ, 1996. - 480с.
3. Кэмпбелл М. Access/ Ответы. - М.: Восточная книжная компания, 1996. - 336с.