

Составитель Л.П. Андреева

Редактор В.В. Чердынцев

Методические указания предназначены для студентов, изучающих дисциплину «Базы данных». Они содержат логически связанную последовательность заданий по разработке приложения баз данных в СУБД ACCESS. Варианты заданий обеспечивают индивидуальную работу каждого студента.

Автор выражает благодарность студентам групп ИП-99 МИРЭА за тестирование методических указаний.

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

Рецензенты: Ф.С. Ильинский, Е.П. Храмцова

© Московский государственный институт радиотехники, электроники и автоматики (технический университет, 2002

Литературный редактор Л.В. Омелянович

Изд. лицензия № 020456 от 04.03.97

Подписано в печать 17.01.2002. Формат 60x84 1/16.

Бумага офсетная. Печать офсетная.

Усл. печ.л. 1,86. Усл.кр. - отг. 7,44. Уч.-изд.л. 2,0.

Тираж 100 экз. Заказ 44. Бесплатно.

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

117454, Москва, просп. Вернадского, 78

## ЛАБОРАТОРНАЯ РАБОТА №1

### ВВОД ДАННЫХ С ПОМОЩЬЮ ФОРМЫ

#### Постановка задачи

Разработать форму для ввода новых записей в базу данных. Форма для ввода заказов приведена на рис. 1.

Варианты задания приведены в табл. 1.

Рис. 1. Форма для ввода нового заказа

#### Методические указания

1. Разработайте базу данных для предметной области, варианты которой приведены в Приложении.
2. Разработайте форму для ввода новых записей в таблицу, указанную в вашем варианте. В качестве источника записей для формы ввода укажите эту таблицу. Воспользуйтесь инструментом *Автоформа* или *Мастер формы* для создания формы, а затем в режиме *Конструктор* настройте форму: измените подпись формы; запретите кнопки перехода, область выделения записи, разделительные линии, кнопку оконного меню; установите тип границы *Окно диалога*; установите для формы значения свойств *Разрешить изменение*, *Разрешить удаление*, *Разрешить добавление*.

ние и Ввод данных таким образом, чтобы форма могла использоваться только для ввода данных. Форма ввода не должна быть большой по размеру. Используйте в форме элементы *Поле со списком* для ввода значений из заранее определенного списка значений. Убедитесь в правильности обхода элементов формы по клавише *Enter (Tab)*. Изменить порядок обхода можно с помощью команды *Последовательность перехода* меню *Вид*.

3. Разместите две кнопки в форме ввода. Для того чтобы форма при нажатии клавиши *Enter* в последнем поле выводила новую запись, разместите кнопки в приложении формы.

4. Напишите процедуры обработки события для кнопок.

Обработчик события «Нажатие кнопки *Зарегистрировать*»

- Закрыть форму

Обработчик события «Нажатие кнопки *Отмена*»

- Если данные в форме были изменены
- Отменить изменения
- Закрыть форму

Методы объекта *DoCmd* и свойства формы, полезные для написания процедур обработки событий, приведены в табл. 2-3. Подробную информацию о свойствах и методах объектов ACCESS VBA можно найти в контекстной справке СУБД ACCESS.

Таблица 1

Варианты задания

№	Таблица	Поля таблицы
1	Работники	Номер работника, фамилия, имя, отчество, дата рождения, адрес, дата регистрации
2	Работники	Табельный номер, фамилия, имя, отчество, адрес, телефон, номер отдела
3	Студенты	Учетный номер, фамилия, имя, отчество, код специальности
4	Книги	Шифр, авторы, наименование, год издания
5	Филиалы фирмы	Код филиала, наименование, адрес, телефон
6	Вузы	Код вуза, наименование, адрес, телефон

7	Расписание	Шифр группы, код дисциплины, дата, время, аудитория, преподаватель
8	Преподаватели	Учетный номер, фамилия, имя, отчество, код кафедры
9	Поставщики	Код поставщика, наименование, адрес, телефон
10	Студенты	Шифр, фамилия, имя, отчество, группа
11	Поставщики	Код поставщика, наименование, адрес, телефон
12	Учебные планы	Код факультета, код специальности, группа, код дисциплины, семестр, количество часов лекций, лр, из, из, вид отчетности в конце семестра
13	Предприятия	Код предприятия, наименование, адрес, телефон
14	Заказы	Номер заказа, ФИО, адрес, дата регистрации, количество, цена экземпляра
15	Заказы	Номер заказа, ФИО, адрес, дата регистрации, количество, сумма оплаты

Таблица 2

Методы объекта *DoCmd*

Метод	Описание	Аргументы
Close	Закрытие объекта	Для текущего объекта отсутствуют
OpenForm	Открытие формы	Имя формы ..., критерий отбора
SetWarnings	Вывод/отмена системных сообщений	True/False
RunSQL	Выполнение команды SQL	Строка SQL-команды
OpenQuery	Выполнение сохраненного запроса	Имя запроса

RunCommand	Выполнение команды встроенного меню	AcCmdDeleteRecord – для удаления, AcCmdUndo – для отмены
GoToRecord	Перемещение к записи	См. справку
SelectObject	Активизация объекта	AcForm, имя формы
GoToControl	Перевод фокуса на элемент управления	Имя элемента управления

Таблица 3

## Свойства формы

Свойство	Описание	Значения
Visible	Вывод на экран	True/False
AllowDeletions	Разрешение удаления записей через форму	True/False
Dirty	Наличие изменений текущей записи	True/False
RecordSource	Источник записей	Имя таблицы или запроса

## ЛАБОРАТОРНАЯ РАБОТА №2

## РЕДАКТИРОВАНИЕ ДАННЫХ

## Постановка задачи

Разработать приложение для просмотра, изменения, ввода и удаления данных. Интерфейс приложения должен содержать три формы: ленточную форму для просмотра и выбора записей, форму для ввода новых данных и форму для изменения выбранной записи. Вид форм для просмотра, ввода и изменения заказов приведен на рис. 1-3. Диалоговое окно для подтверждения удаления заказа приведено на рис.4. Интерфейс приложения должен обеспечивать в любой момент времени вывод на экран только одной формы. Варианты задания приведены в табл. 1.

Имя записи				Сум.	Дата
№ п/п	Дата заказа	Комп.	Адрес		
42	01.02.01	Машину В.В.	121496 г. Москва ул. Мухоморова 24/5	211	300.000
43	01.02.01	Платон П.П.	345678 Ереван обл. Петрозавкандский р-н. Петрозавканд	4	300.000
44	01.02.01	Васильев Г.А.	145789 Тверская обл. Осташковский р-н г. Осташков ул. Л.	3	300.000
45	14.02.01	Дубинин В.В.	127403 г. Москва ул. Пашковская в.В. корп. 2 кв.50	4	300.000

Рис. 2. Ленточная форма для просмотра заказов

Свойства записи	
номер заказа	44
Комп.	Васильев Г.А.
Индекс	345789
Область	Тверская обл.
Район	Осташковский р-н
Корrespondент	г. Осташково
Адрес	ул. Советская в. 34
Количество	2
Цена заказа	300.000
Дата заказа	01.02.01

Рис. 3. Форма для изменения заказа

## Методические указания

1. Разработайте форму для изменения выбранной записи. Разместите в ней две кнопки. Напишите процедуры обработки событий для этих кнопок.

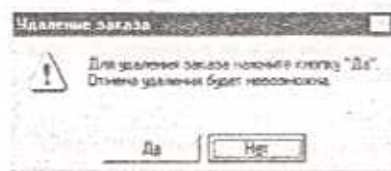


Рис. 4. Окно подтверждения удаления заказа

2. Разработайте ленточную форму. В качестве источника записей для этой формы укажите таблицу или многотабличный запрос на основе таблиц вашей базы данных. Для предотвращения непреднамеренного удаления или изменения записи установите для формы режим, запрещающий удаление и изменение записей через форму. Разместите в форме 4 кнопки. Напишите для этих кнопок процедуры обработки событий.

Обработчик события «Нажатие кнопки *Добавить*»

- Перевести ленточную форму в скрытое состояние
- Открыть форму для ввода новой записи

Обработчик события «Нажатие кнопки *Изменить*»

- Перевести ленточную форму в скрытое состояние
- Открыть форму для изменения выбранной записи.

Обратите внимание на синтаксис задания критерия отбора записи в методе открытия формы объекта *DoCmd* для строковых полей и полей типа «Дата». Ниже приведены примеры вызова метода открытия формы объекта *DoCmd*.

```
DoCmd.OpenForm "Изменение", , "ФИО= 'Иванов И.И. '"
DoCmd.OpenForm "Изменение", , "ФИО= '" & Me!ФИО & "'"
DoCmd.OpenForm "Изменение", , "ФИО= Forms!Просмотр!ФИО "
DoCmd.OpenForm "Изменение", , "[Дата рождения]=#01/02/99#"
DoCmd.OpenForm "Изменение", , "[Дата рождения]=#" &
Month(Me!Дата рождения) & "/" & Day(Me!Дата рождения)) &
"/" & Year(Me!Дата рождения) & "#"
```

Обработчик события «Нажатие кнопки *Удалить*»

- Вывести диалоговое окно подтверждения удаления. Для вывода окна используйте функцию *MsgBox*. Пример использования этой функции в процедуре обработки события «Нажатие кнопки *Удаление* приведен» ниже.

```
Private Sub Удаление_Click()
```

```
If MsgBox("Для удаления заказа нажмите кнопку ДА. " &
Chr(10) & "Отмена удаления будет невозможна.", _
VbYesNo+VbExclamation+VbDefaultButton2, "Удаление заказа") _
= VbYes Then
    DoCmd.RunSQL "delete from [заказы] where [код заказа]=#" &
    Me![Код заказа]
    Me.Requery
```

```
End If
```

```
End Sub
```

- Если пользователь подтверждает удаление
- Отменить вывод системных сообщений
- Удалить запись
- Обновить ленточную форму (метод формы *Requery*)
- Восстановить вывод системных сообщений

Обработчик события «Нажатие кнопки *Выход*»

- Закрыть ленточную форму

3. Напишите обработчик события «Закрытие формы» для формы ввода записей.

- Перевести ленточную форму в видимое состояние
- Обновить ленточную форму

4. Напишите обработчик события «Закрытие формы» для формы изменения записей.

5. Максимизируйте размер ленточной формы. Для этого используйте метод *Maximize* объекта *DoCmd*. Установите для форм ввода и изменения данных свойство *Всплывающее окно* в значение «Да». Свойство *Всплывающее окно*, установленное в значение «Да», обеспечивает возможность сохранить свои размеры формам приложения, которые не должны разворачиваться до максимального размера.

## ЛАБОРАТОРНАЯ РАБОТА №3

### ПОИСК ГРУППЫ ЗАПИСЕЙ

#### Постановка задачи

Разработать форму для поиска группы записей по заданным критериям. Пример формы для поиска заказов приведен на рис. 5. Форма содержит поля для задания условий отбора записей, список, в котором отображаются записи, отвечающие условиям поиска, и командные кнопки. Кнопка *Поиск* выполняет поиск записей и вывод в список найденных записей. Кнопка *Сброс* сбрасывает текущие условия поиска, заполняя поля пустыми значениями, и выводит все записи. Варианты задания приведены в табл. 4.

Имя	Сумма	Отгрузка	Вид
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
42 Иванку В.В.	123456 г. Москва ул. Петровка, 24-5	300.000	14.02.01
43 Петров П.П.	345678 Карские Альпы, Петровский район, Петровка	300.000	
44 Батильярой Т.А.	345789 Тьерра-де-Лос-Анджелес-де-Орландо, ул. С	300.000	
45 Аткинсу В.В.	123456 г. Москва ул. Петровка, в.б. этаж 2, кв.50	300.000	

Рис. 5. Форма для поиска заказов

#### Методические указания

1. Создайте запрос для вывода всех записей. Он будет являться источником строк для списка (свойство списка *RowSource*) при

пустых значениях полей, задающих условия отбора записей. Создайте параметрический запрос, для вывода записей, удовлетворяющих условиям поиска. В выражениях для условий отбора задайте ссылки на поля формы поиска. Для поиска по первым символам строкового значения поля используйте оператор *Like* или функции *Left* и *Len*. Параметрический запрос будет использоваться в качестве источника записей списка при поиске записей с помощью формы. Можно разработать одно параметрическое условие для списка и в нем предусмотреть условие для вывода всех записей при пустых значениях полей, задающих условия отбора записей. Текст *SQL* запроса для поиска заказов для формы «Поиск заказов без предоплаты», приведен на рис. 6.

2. Разработайте несвязанную с источником данных форму для поиска информации.
3. Разместите в форме несвязанные с источниками данных поля для ввода условий поиска. Если для поиска используются даты, то установите для полей формат *дд.мм.гггг*. Для задания критерия отбора можно использовать поле со списком, если количество возможных значений критерия невелико.
4. Разместите в форме список, указав для него в качестве источника строк имя запроса.
5. Разместите в форме кнопки *Поиск*, *Сброс* и *Выход*. Напишите для этих кнопок обработчики событий. Используйте для обновления списка метод *Requery* или свойство списка *RowSource*. Включите в обработчик события «Нажатие кнопки *Поиск*» проверку заполнения всех полей, если условия отбора связаны логической операцией «И». Включите в обработчик события «Нажатие кнопки *Сброс*» очистку полей ввода условий.
6. Напишите обработчик события «Открытие формы» для установки фокуса на первое поле условия отбора.

Таблица 4

#### Варианты задания

№	Информация, выводимая в списке	Критерий поиска
1	Список работников (Фамилия и инициалы, дата рождения, адрес)	Диапазон дат рождения, наименование

		профессии
2	Список продавцов, продающих данный товар=(Наименование товара, фамилия и инициалы, номер отдела)	Первые буквы наименования товара
3	Список студентов=(Учетный номер, фамилия и инициалы)	Семестр, специальность, наименование дисциплины
4	Список книг=(Шифр, авторы, наименование)	Первые буквы авторов, первые буквы наименования
5	Список товаров=(Наименование филиала, вид товара, наименование поставщика)	Первые символы наименования филиала
6	Список выпускников = (Наименование вуза, наименование специальности, фамилия и инициалы, средний балл)	Наименование специальности, наименование вуза
7	Расписание экзаменов преподавателя=(Шифр группы, время, аудитория)	Преподаватель, дата
8	Список преподавателей = (Наименование дисциплины, фамилия и инициалы преподавателя, наименование кафедры)	Первые буквы наименования дисциплины
9	Список партий товаров=(Вид товара, наименование поставщика, дата закупки, закупочная цена, количество)	Вид товара, наименование поставщика
10	Список студентов, получивших хотя бы одну оценку «2» =(Фамилия, имя, отчество)	Группа, семестр, дисциплина
11	Список поставщиков=(Наименование детали, наименование поставщика, адрес, телефон)	Первые символы наименования детали
12	Учебные планы=(дисциплина, вид	Наименование фа-

	отчетности, количества часов)	культета, наименование специальности, семестр
13	Список вакансий=(Наименование профессии, наименование предприятия, адрес)	Первые символы наименования профессии
14	Список заказов=(Номер заказа, фамилия и инициалы, адрес, дата регистрации, количество)	Первые буквы фамилии, первые символы адреса
15	Список заказов=(Номер заказа, фамилия и инициалы, адрес, дата регистрации, количество)	Диапазон дат регистрации заказа

#### ЛАБОРАТОРНАЯ РАБОТА №4

#### ИСПОЛЬЗОВАНИЕ СПИСКА С МНОЖЕСТВЕННЫМ ВЫБОРОМ ДЛЯ ПЕРЕМЕЩЕНИЯ ЗАПИСЕЙ

##### Постановка задачи

Разработать форму для перемещения произвольно выбранных пользователем записей из основной таблицы в архивную таблицу. Для произвольного выбора записей использовать список с множественным выбором. Пример формы, в которой используются два списка с множественным выбором строк, приведен на рис. 7. Кнопки между списками предназначены для перемещения выбранной информации из одного списка в другой. Кнопка *Сформировать* предназначена для выполнения операции формирования почтового списка из отображенных записей. Варианты основной таблицы приведены в табл. 1.

##### Методические указания

1. Создайте архивную таблицу «АРХИВ», скопировав для нее структуру основной таблицы вашей базы данных.
2. Установите в схеме данных каскадное удаление записей подчиненных таблиц, связанных с основной таблицей.

3. Создайте таблицу «РАБОЧАЯ» для временного хранения информации о перемещаемых записях. В структуру этой таблицы включите ключевые поля основной таблицы. Можно включить и другие поля основной таблицы, которые должны отображаться в правом списке.

4. Разработайте запрос для поиска записей основной таблицы, значения ключевых полей которых не совпадают со значениями ключевых полей записей таблицы «РАБОЧАЯ». Для этого используйте в запросе подчиненный запрос или левое внешнее объединение (*Left Join*). Предусмотрите в запросе вывод полей, которые должны отображаться в списке. Разработанный запрос будет использоваться в качестве источника строк левого списка.



Рис. 7. Форма для формирования списков для почты

5. Разработайте форму для перемещения записей.

5.1. Разместите в форме левый список, указав для него в качестве источника строк имя запроса, разработанного в пункте 4. Для множественного выделения строк в списке установите значение свойства списка *Несвязанное выделение* в значение «Простой». Выберите в качестве присоединенного столбца списка ключевой

столбец основной таблицы, установив в свойстве *Присоединенный столбец* номер этого столбца. Задайте ширину столбцов списка. Протестируйте форму. Убедитесь, что в левом списке можно выделить несколько строк.

5.2. Разместите в форме правый список, указав для него в качестве источника строк таблицу «РАБОЧАЯ». Для множественного выделения строк в списке установите значение свойства списка *Несвязанное выделение* в значение «Простой». Выберите в качестве присоединенного столбца списка ключевой столбец таблицы.

5.3. Разместите 4 кнопки между списками и напишите обработчики событий для этих кнопок.

Обработчик события «Нажатие кнопки >» должен обеспечивать перемещение отобранных в левом списке записей в правый список:

- Если левый список пустой
  - Вывести сообщение
  - Завершить выполнение процедуры
- Если в левом списке нет выделенных строк
  - Вывести сообщение
  - Завершить выполнение процедуры
- Цикл по выделенным строкам левого списка
  - Добавить в таблицу «РАБОЧАЯ» информацию о выделенной записи основной таблицы
  - Отменить выделение строки
- Конец цикла
- Обновить левый список
- Обновить правый список

При программировании операций добавления или удаления записей таблиц используйте методы *RunSQL* или *OpenQuery* объекта *DoCmd*. Пример процедуры обработки события «Нажатие кнопки >» для формы формирования списков заказов для почты приведен на рис. 8.

Обработчик события «Нажатие кнопки <» должен перемещать отобранные в правом списке записи в левый список. Обра-

ботчик события «Нажатие кнопки >>» должен перемещать все записи левого списка в правый список. Обработчик события «Нажатие кнопки <<» должен перемещать все записи правого списка в левый список.

5.4. Разместите кнопку *Выполнить*. Напишите обработчик события «Нажатие кнопки *Выполнить*». Он должен перемещать отобранные из основной таблицы записи, которые отображаются в правом списке, в таблицу «АРХИВ» и отображать в форме результат выполнения архивации:

- Добавить в таблицу «АРХИВ» все отобранные записи
- Удалить из основной таблицы все отобранные записи
- Очистить таблицу «РАБОЧАЯ»
- Обновить левый список
- Обновить правый список

5.5. Разместите в форме кнопку *Выход*. Напишите обработчик события «Нажатие кнопки *Выход*». Он должен обеспечивать очистку таблицы «РАБОЧАЯ», восстановление вывода системных сообщений и закрытие формы.

5.6. Напишите обработчик события «Открытие формы», который должен отменять вывод системных сообщений при выполнении запросов действия.

Краткое описание свойств списка, полезных при написании обработчиков событий для разрабатываемой формы, приведено в табл. 5.

Таблица 5

Свойства списка

Свойство	Описание
Selected (i)	Состояние выделения i-строки списка (True/False)
ItemData(i)	Значение, содержащееся в присоединенном столбце i-строки списка
Column(j, i)	Значение элемента списка, расположенного в j-столбце и i-строке (нумерация строк и столбцов начинается с 0)

RowSource	Источник строк списка (имя таблицы, запроса или SQL-команда)
ItemsSelected	Семейство, содержащее индексы выделенных строк. Имеет свойство Count – число элементов в семействе (количество выделенных строк списка)
ColumnCount	Количество столбцов в списке
ListCount	Количество строк в списке
Count	Количество элементов в семействе ItemsSelected (количество выделенных строк списка)

```
Private Sub Вправо_Click()
Dim i
If Список1.ListCount = 0 Then
MsgBox "Список пустой"
Exit Sub
End If
If Список1.ItemsSelected.Count = 0 Then
MsgBox "Нет выделенных строк"
Exit Sub
End If
For Each i In Список1.ItemsSelected
DoCmd.RunSQL "insert into [Рабочая]([Код заказа] , " & _
"[Количество]) select [Код заказа],[Количество]from [Заказы]" _
& "where [Код заказа]= " & Список1.ItemData(i)
Список1.Selected(i) = False
Next
Список1.Requery
Список2.Requery
End Sub
```

Рис. 8. Текст процедуры для перемещения записей



## ЛАБОРАТОРНАЯ РАБОТА №5

### АВТОМАТИЗАЦИЯ ПОДГОТОВКИ И ПЕЧАТИ ОТЧЕТА

#### Постановка задачи

Разработать интерфейс для просмотра и печати отчета. Пример диалоговой формы, автоматизирующей работу с отчетом по возврату заказов на определенную дату, приведен на рис. 9. Кнопка *Просмотр* предназначена для вывода отчета в режиме просмотра, кнопка *Печать* – для печати отчета. Пользователь должен иметь возможность подтвердить печать отчета и указать количество экземпляров отчета для печати. При отсутствии данных для отчета должно быть выведено соответствующее информационное сообщение. Заголовок отчета должен содержать значения введенных параметров. Так, отчет о возврате заказов на 2.05.01 должен иметь заголовок: «Отчет по возврату заказов на 2.05.01». Варианты задания приведены в табл.6.

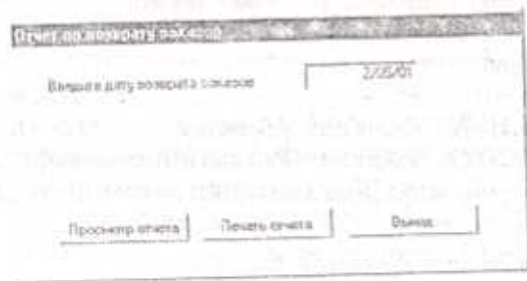


Рис. 9. Диалоговая форма для ввода параметра отчета

#### Методические указания

1. Создайте запрос, который будет использоваться в качестве источника данных для отчета. В выражениях для условий отбора укажите ссылки на поля формы. Группировку записей в запрос можно не включать, так как ее можно будет выполнить при создании макета отчета с помощью мастера. Пример задания для отчета по возврату заказов на заданную дату, приведен на рис. 10.

2. Разработайте макет отчета, указав для него в качестве источника записей имя разработанного запроса. Макет отчета можно создать с помощью мастера отчета, а затем откорректировать его в конструкторе. Укажите мастеру поля, по которым будет выполняться группировка и интервал группировки, например, по месяцам. Сформируйте в области заголовка отчета заголовок отчета, включающий значения параметров отчета. Добавьте к макету отчета примечание группы, если вариантом задания предусмотрен вывод итоговых значений по группе (команда *Сортировка и группировка* в меню *Вид*). Разместите в примечании группы итоговые поля, указав в них статистические функции. Например, для вывода количества возвращенных заказов укажите `=Count([Код заказа])`. В примечание отчета можно перетащить из заголовка группы выражение для вычисления месяца, если вывод месяца предусмотрен вариантом. Если в отчете надо выводить только итоговые значения по группе, то установите для области данных свойство *Вывод на экран* в значение `«Нет»`.

3. Разработайте диалоговую форму для ввода параметров отчета. Разместите в диалоговой форме три кнопки и напишите для них обработчики событий.

#### Обработчик события «Нажатие кнопки *Просмотр*»

- Проверить наличие данных для отчета (используйте функцию *Dcount*)
- Если данные для отчета отсутствуют
- Вывести соответствующее сообщение. Пример использования функций *Dcount* и *MsgBox* для вывода информационного сообщения для отчета о возврате заказов приведен на рис.11.
- Закончить процедуру
- Вывести отчет в режиме просмотра

Обработчик события «Нажатие кнопки *Печать*» должен проверить наличие данных для отчета, вывести окно подтверждения печати, ввести количество экземпляров отчета (по умолчанию это количество должно быть равным 1) и выводить печать отчета.

Таблица 6

## Варианты задания

№	Содержание отчета	Параметры отчета	Выводимые поля	Дополнительные требования
1	Отчет за период о регистрации безработных	Начало и конец периода	Месяц количество зарегистрированных за месяц безработных	Группировка по месяцам, итоговое количество за период
2	Информация о ценах на товар	Наименование товара	Цены изготовителей на товар с указанием наименований и адресов фирм изготовителей	Сортировка по наименованиям фирм, минимальная, максимальная и средняя цена товара
3	Учебные планы по дисциплинам для заданной специальности	Специальность и семестр	Наименования дисциплин, количества часов лекций, лабораторных, индивидуальных, практических занятий и отчетность	Сортировка по дисциплинам
4	Отчет за период о работе абонемента	Начало и конец периода	Месяц количество выполненных операций сдачи/выдачи книг в месяц	Группировка по месяцам, итоговое количество за период
5	Отчет о поступлении товаров в филиал	Наименование филиала	Вид и количество товара с указанием поставщика	Группировка по виду товара, итоговое количество по каждому виду товара

6	Информация о выпускниках по заданной специальности	Специальность	Вуз и количество выпускников для каждого вуза	Группировка по вузам, общее количество выпускников по заданной специальности
7	Расписание экзаменов для заданной группы и заданного диапазона дат	Группа, диапазон дат	Дата, время, аудитория, преподаватель	Сортировка по датам
8	Список преподавателей по заданной дисциплине	Дисциплина	Фамилия и инициалы, кафедра, факультет	Группировка по кафедрам, общее количество преподавателей
9	Отчет о закупке товаров за период по месяцам	Начало и конец периода	Месяц, вид товара, количество, сумма по каждому виду товаров	Группировка по виду товаров, итоговое количество и сумма
10	Результаты экзаменов для заданного семестра	Семестр	Группа, дисциплина, средний балл по дисциплине для каждой группы	Группировка по группам и дисциплинам, средний балл по каждой группе
11	Отчет за период о поставке деталей	Начало и конец периода	Вид детали, наименование поставщика, количество деталей	Группировка по видам деталей, итоговое количество деталей по каждому виду деталей
12	Учебные планы фа-	Вид групп-	Наименование дисциплины,	Группировка по группам или

	культета для заданного семестра	пировки	группа, количества часов лекций, лабораторных, индивидуальных, практических занятий и отчетность	группировка по дисциплинам
13	Отчет о вакансиях предприятий	Наименование профессии	Наименование предприятия, суммарное количество вакансий, зарплата по профессии для каждого предприятия	Сортировка по наименованиям предприятий, общее количество вакансий и средняя зарплата по профессии
14	Отчет о зарегистрированных заказах за период по месяцам	Начало и конец периода	Месяц, количество и стоимость зарегистрированных заказов за месяц	Группировка по месяцам, общее количество и стоимость заказов за период
15	Отчет о зарегистрированных заказах за период по месяцам	Начало и конец периода	Месяц, количество зарегистрированных заказов, общая сумма оплаты за месяц	Группировка по месяцам, общее количество заказов и общая сумма оплаты за период

```
Select [Дата возврата],[Код заказа]
From Заказы
Where [Дата возврата] = Forms![Отчет]! Дата
```

Рис. 10. Запрос для отчета по возврату заказов

```
If Dcount("*","Заказы",[Дата возврата] = Forms![Отчет]! Дата) _
=0 Then
MsgBox "Отчет не содержит данных. " & Chr(10)
& "Проверьте правильность даты " , vbOKOnly + vbInformation
```

Рис. 11. Вывод информационного сообщения

## ЛАБОРАТОРНАЯ РАБОТА №6

### СОЗДАНИЕ ИНТЕРФЕЙСА ПРИЛОЖЕНИЯ

#### Постановка задачи

Разработать интерфейс приложения для управления задачами: редактированием данных, поиском информации по заданным критериям, архивацией отобранных пользователем данных и печатью отчета. Интерфейсе должен включать главную кнопочную форму, ленточную форму для просмотра и выбора данных, формы для ввода и изменения данных, форму для поиска информации, форму для архивации, форму для просмотра и печати отчета. Вид главной кнопочной формы для приложения «Наложённые платежи» приведен на рис. 12. Параметры запуска приложения должны обеспечивать автоматическое открытие главной формы.

#### Методические указания

1. Разработайте главную форму приложения. Напишите обработчики событий для кнопок формы и подписей к ним. Нажатие кнопки или подписи к ней должно скрывать главную форму и открывать форму для выполнения выбранной задачи. Нажатие кнопки *Выход* должно закрывать главную форму и выполнять выход из СУБД Access. Для выхода из СУБД используйте метод *Quit* объекта *Application*. Для каждой формы, которая будет вызываться из главной, в обработчике события «Закрытие формы» предусмотрите перевод главной формы в видимое состояние.
2. Установите для всех форм свойства *Модальное окно* и *Всплывающее окно* в значение «Да». Это не позволит пользователю при работе с активной формой получить доступ к другим объектам

	культета для заданного семестра	пировки	группа, количества часов лекций, лабораторных, индивидуальных, практических занятий и отчетность	группировка по дисциплинам
13	Отчет о вакансиях предприятий	Наименование профессии	Наименование предприятия, суммарное количество вакансий, зарплата по профессии для каждого предприятия	Сортировка по наименованиям предприятий, общее количество вакансий и средняя зарплата по профессии
14	Отчет о зарегистрированных заказах за период по месяцам	Начало и конец периода	Месяц, количество и стоимость зарегистрированных заказов за месяц	Группировка по месяцам, общее количество и стоимость заказов за период
15	Отчет о зарегистрированных заказах за период по месяцам	Начало и конец периода	Месяц, количество зарегистрированных заказов, общая сумма оплаты за месяц	Группировка по месяцам, общее количество заказов и общая сумма оплаты за период

```
Select [Дата возврата],[Код заказа]
From Заказы
Where [Дата возврата] = Forms![Отчет]! Дата
```

Рис. 10. Запрос для отчета по возврату заказов

```
If Dcount("*","Заказы",[Дата возврата] = Forms![Отчет]! Дата) = 0 Then
MsgBox "Отчет не содержит данных." & Chr(10)
& "Проверьте правильность даты " , vbOKOnly + vbInformation
```

Рис. 11. Вывод информационного сообщения

## ЛАБОРАТОРНАЯ РАБОТА №6

### СОЗДАНИЕ ИНТЕРФЕЙСА ПРИЛОЖЕНИЯ

#### Постановка задачи

Разработать интерфейс приложения для управления задачами: редактированием данных, поиском информации по заданным критериям, архивацией отобранных пользователем данных и печатью отчета. Интерфейс должен включать главную кнопочную форму, ленточную форму для просмотра и выбора данных, формы для ввода и изменения данных, форму для поиска информации, форму для архивации, форму для просмотра и печати отчета. Вид главной кнопочной формы для приложения «Наложённые платежи» приведен на рис. 12. Параметры запуска приложения должны обеспечивать автоматическое открытие главной формы.

#### Методические указания

1. Разработайте главную форму приложения. Напишите обработчики событий для кнопок формы и подписей к ним. Нажатие кнопки или подписи к ней должно скрывать главную форму и открывать форму для выполнения выбранной задачи. Нажатие кнопки *Выход* должно закрывать главную форму и выполнять выход из СУБД Access. Для выхода из СУБД используйте метод *Quit* объекта *Application*. Для каждой формы, которая будет вызываться из главной, в обработчике события «Закрытие формы» предусмотрите перевод главной формы в видимое состояние.
2. Установите для всех форм свойства *Модальное окно* и *Всплывающее окно* в значение «Да». Это не позволит пользователю при работе с активной формой получить доступ к другим объектам

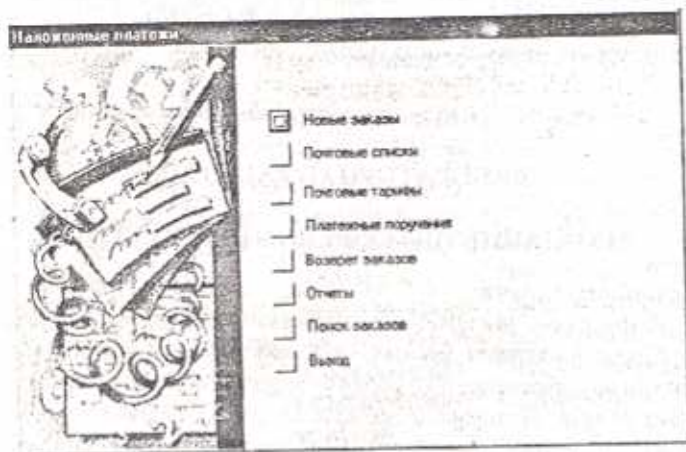


Рис. 12. Главная кнопочная форма приложения «Наложенные платежи»

приложения. Свойство *Всплывающее окно*, установленное в значение «Да», обеспечивает также возможность сохранить свои размеры формам приложения, которые не должны разворачиваться до максимального размера.

3. Максимизируйте размер отчета для просмотра.

4. Установите параметры запуска приложения (*Меню, Сервис, Параметры запуска*):

- Укажите заголовок приложения
- Отмените вывод окна базы данных, набора настроек меню, контекстного меню и стандартных панелей инструментов
- Укажите форму, которая будет открываться автоматически при запуске приложения.

5. После установок параметров запуска перезагрузите приложение.

## ЗАДАНИЕ №7

### РАБОТА С НАБОРАМИ ЗАПИСЕЙ

#### Постановка задачи

Разработать процедуру, добавляющую в таблицу новые записи, сформированные из данных других таблиц. При программировании процедуры использовать наборы записей (объекты *RecordSet*).

Варианты задания приведены в табл. 7.

#### Методические указания

1. Разработайте структуру таблицы «Списки».
2. Создайте простую форму с кнопкой для вызова процедуры добавления новых записей в таблицу «Списки».
3. Создайте запрос, если в варианте для входного набора записей используется запрос, предусмотрен в нем сортировку данных.
4. Напишите процедуру обработки события «Нажатие кнопки».

Пример процедуры поиска в таблице «Заказы» заказа с максимальным количеством экземпляров товара приведен на рис. 13.

Таблица 7

#### Варианты задания

№	Правило формирования записей для таблицы «Списки»	Структура таблицы «Списки»
1	Сформировать списки работников, имеющих одну, две и т.д. профессий. Упорядочить списки по количеству профессий	Номер списка (равной количеству профессий), номер элемента списка, номер работника
2	Сформировать списки работников для каждого отдела	Номер отдела, номер элемента списка, табельный номер
3	Сформировать списки студентов для посещения занятий по заданной дисциплине по заданной специальности	Номер списка, номер элемента списка, учетный номер

	ности в заданном семестре. Количество студентов в списке задает пользователь	
4	Сформировать список книг автора (авторов), имеющего в библиотеке максимальное количество книг	Номер элемента списка, название книги, год издания
5	Сформировать список поставщиков, поставляющих товары в каждый филиал торговой фирмы	Номер элемента списка, наименование поставщика, адрес поставщика
6	Сформировать списки выпускников по каждой специальности, включив в каждый список не более n выпускников, имеющих наибольший средний балл	Номер списка, специальность, вуз, фамилия, имя, отчество, средний балл
7	Сформировать для каждой даты из расписания экзаменов список преподавателей принимающих экзамены	Дата, номер элемента списка, фамилия, имя, отчество
8	Сформировать списки преподавателей по каждой дисциплине	Номер списка, код дисциплины, учетный номер преподавателя, кафедра
9	Сформировать по каждому виду товара список поставщиков, поставляющих данный вид товара	Номер списка, номер элемента списка, вид товара, код поставщика
10	Списки неуспевающих студентов в заданном семестре: в список №1 включить студентов, имеющих три и более оценки «2», в список №2 – две «2», в список №3 – одну «2»	Номер списка, фамилия, имя, отчество, группа
11	Сформировать списки поставщиков для каждого вида деталей	Номер списка, номер элемента списка, код вида, наименование

		поставщика.
12	Сформировать список дисциплин с максимальным количеством часов, отводимых на изучение дисциплины	Номер элемента списка, наименование дисциплины, код специальности, факультет
13	Сформировать список специальностей, по которым есть вакансии только на одном предприятии	Номер элемента списка, наименование специальности, наименование предприятия
14	Сформировать список заказов для почты, включив в каждый элемент списка по одному экземпляру заказа. Количество элементов в списке задает пользователь	Номер списка, номер элемента списка, код заказа, ФИО, адрес
15	Сформировать список заказов, включив в каждый элемент списка по одному заказу. Количество элементов в списке задает пользователь	Номер списка, номер элемента списка, код заказа, ФИО, адрес

```

Private Sub Кнопка0_Click()
Dim max As Integer 'максимальное количество экземпляров
Dim intmax As Variant 'указатель на запись
Dim r1 As Recordset 'набор записей
Dim db As Database 'текущая база данных
Set db = CurrentDb
Set r1 = db.OpenRecordset("Заказы") 'открытие набора записей
If r1.RecordCount = 0 Then
MsgBox "Нет записей"
r1.Close
Exit Sub
End If
r1.Index="PrimaryKey" 'Порядок перемещения по записям
max = 0
Do While Not r1.EOF

```

```

If r1![экземпляры] > max Then
  max = r1![экземпляры]
  intmax = r1.Bookmark 'сохранение адреса текущей записи
End If
r1.MoveNext           'переход к следующей записи
Loop
' Переход к записи с максимальным количеством экземпляров
r1.Bookmark = intmax
MsgBox r1![Код заказа] & " " & r1![Фино]& " " & r1![экземпляры]
r1.Close
End Sub

```

Рис. 13. Текст процедуры поиска максимального заказа

## ПРИЛОЖЕНИЕ

### ВАРИАНТЫ ПРИЛОЖЕНИЙ БАЗ ДАННЫХ

#### 1. Служба занятости

Приложение учитывает информацию о предложениях работников. Информация о предложении работника: регистрационный номер работника, фамилия, имя, отчество, дата рождения, адрес, список профессий, которыми владеет работник (код профессии, наименование профессии, наличие опыта работы по профессии).

#### 2. Магазины

Приложение отслеживает распределение товаров по отделам магазина. Магазин имеет отделы, в которых продаются товары. Информация о каждом отделе содержит номер отдела, название 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

Приложение ведет учет изучаемых студентами дисциплин. 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 дисциплин составляется для каждой специальности по каждому семестру. Для каждого элемента списка указываются дисциплина, специальность, номер семестра, количеств часов в неделю лекций, практических занятий, лабораторных работ и вид отчетности в конце семестра. Специальность имеет код и наименование. Дисциплина имеет код и наименование.

#### 4. Библиотека

Приложение автоматизирует работу абонента библиотеки. Читатели библиотеки получают на время книги в библиотеке. Каждая операция выдачи/сдачи 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

Приложение ведет учет товаров в филиалах торговой фирмы. Филиалы торговой фирмы, имеющие наименование, адрес и телефон, продают различные виды товаров. Для каждого филиала ведется учет текущего количества каждого вида товаров. Товар в филиалах 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

Приложение автоматизирует работу службы занятости. В службе занятости о каждом вузе имеется следующая информа-

ция: наименование вуза, адрес, телефон, список выпускников. Каждый выпускник имеет фамилию, имя, отчество, специальность, средний балл. О каждой фирме, 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

Приложение автоматизирует работу с расписанием 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. Картотека преподавателей

Приложение автоматизирует делопроизводство отдела кадров вуза. В картотеке 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. Преподаватели института могут работать только на одной кафедре. Кафедра принадлежит одному факультету. Факультет имеет код, наименование и список принадлежащих ему кафедр. Кафедра имеет код и наименование.

#### 9. Оптовая фирма

Приложение ведет учет товаров, закупаемых у поставщиков. Фирма закупает партии товаров у поставщиков. Информация о каждой партии товара: вид товара, поставщик, закупочная цена, количество, дата закупки. Поставщик имеет наименование, адрес, телефон и может поставлять несколько видов товара.

#### 10. Результаты экзаменационной сессии

Приложение автоматизирует работу деканатов вуза. Студенты каждый семестр сдают экзамены по нескольким дисциплинам. Студенты могут пересдавать экзамен несколько раз разным преподавателям. В один день студент не может сдавать несколько экзаменов. Информация о студенте: шифр, фамилия, имя, отчество, группа. Информация о каждом экзамене: наименование дисциплины, код дисциплины, семестр, дата сдачи, преподаватель, оценка.

#### 11. Поставка деталей

Приложение ведет учет поставляемых на завод 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

Приложение автоматизирует работу учебного управления вуза, связанную с учебными планами. 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.

#### 13. Служба занятости

Приложение автоматизирует работу службы занятости. Информация о вакансии предприятия: код предприятия, наименование предприятия, адрес, муниципальный округ (код муниципаль-



ного 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).

#### 14. Наложённые платежи

Приложение автоматизирует работу, фирму с заказами. При регистрации заказа в базу данных записывается информация о заказе, заказчике и о работнике фирмы, который принял заказ. Информация о заказе: номер заказа, дата регистрации заказа, количество экземпляров товара в заказе, цена экземпляра товара на день регистрации товара. Информация о заказчике: фамилия и инициалы заказчика, адрес заказчика. Информация о работнике: фамилия и инициалы работника, должность.

#### 15. Наложённые платежи

Приложение автоматизирует работу, фирму с заказами с предоплатой. При регистрации заказа в базу данных записывается информация о заказе, заказчике и о работнике фирмы, который принял заказ. Информация о заказе: номер заказа, дата регистрации заказа (дата предоплаты), количество экземпляров товара в заказе, сумма предоплаты всего заказа. Информация о заказчике: фамилия и инициалы заказчика, адрес заказчика. Информация о работнике: фамилия и инициалы работника, должность.

### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Новалис С. Access 97. Руководство по макроязыку и VBA. – М.: Лорд, 1998. – 590с.
2. Андреева Л.П. Базы данных. Методические указания по выполнению лабораторных работ. – М.: МИРЭА, 2000. – 32с.
3. Кирсанов Д. Веб-дизайн: книга Дмитрия Кирсанова. – СПб: Символ-Плюс, 2001 – 376с.