

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РФ
ФГОУ ВПО «САНКТ – ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
АГРАРНЫЙ УНИВЕРСИТЕТ»

Кафедра ВТ и ИО АПК

Работа с базами данных в MS ACCESS

Методическое пособие
по дисциплине «Информатика»

Санкт – Петербург
2008

Составители: профессор, д.т.н. В.В. Белов
профессор, д.т.н. О.Г. Огнев
доцент. Н.Н. Белова
доцент, к.э.н. А.Л. Попова
доцент, к.э.н. И.Н. Якушева

В методическом пособии кратко изложены: основные понятия Microsoft Access, задания к лабораторным работам и методика их выполнения.

Рекомендовано методической комиссией экономического факультета СПбГАУ (протокол № 2 от 18 января 2008 г.) в качестве методического пособия по дисциплине «Информатика» для студентов всех специальностей.

Рецензенты: профессор, к.э.н. А.П. Савина
профессор, к.т.н. И.С. Давыдов

Предисловие

Это методическое пособие предназначено для изучения СУБД (систем управления базами данных) на примере программы ACCESS, входящей в стандартный пакет Microsoft Office. Пособие содержит краткие теоретические сведения и курс лабораторных работ, который желательно проходить наряду с лекционным преподаванием.

Основная цель авторов – дать знания и укрепить навыки, достаточные для самостоятельного изучения СУБД.

Для начала работы с данным пособием необходимо знать и уметь:

- ✓ основные приемы работы с операционной системой Windows;
- ✓ основные приемы работы с текстовым редактором MS-Word;
- ✓ основные приемы работы с табличным процессором MS-Excel;
- ✓ основные приемы работы с графическими объектами Microsoft Office;
- ✓ уверенно использовать манипулятор «мышь»;
- ✓ работа с контекстными меню.

Введение

В самом общем смысле **база данных** – это один или несколько файлов, содержащих связанную между собой по смыслу информацию. Допустим, в одном файле хранятся даты рождения ваших друзей, в другом - ваши совместные фотографии. Или можно создать несколько файлов электронных таблиц, где учитываются ваши доходы и расходы за определенные периоды. Если Вы организованный человек, то, используя специальную структуру каталогов и подкаталогов, Вы справитесь с сотней - другой файлов. Но, рано или поздно, Вы столкнетесь с такой задачей, которая будет слишком велика. Что делать? Что делать, если какие-то данные должны использовать несколько приложений или людей, и при этом нельзя допустить, чтобы одни и те же данные корректировались одновременно? Когда Вы оказываетесь перед подобными проблемами, Вам нужна **система управления базами данных** (СУБД).

Почти все современные системы основаны на реляционной (relational) модели управления базами данных. Название «реляционная» связано с тем, что каждая запись содержит информацию, относящуюся только к одному объекту. В таких базах данные не дублируются, а связываются по определенным полям. Например, при указании товаров и их категорий нецелесообразно упоминать все сведения о поставщике, которые хранятся в отдельной таблице. Достаточно связать данные о товарах с выше указанной таблицей по номеру (коду) поставщика.

Можно выделить три основные функции СУБД:

- ◆ **определение данных** (Data definition) – можно определить, какая именно информация будет храниться в базе данных, задать структуру данных и их тип (например, максимальное количество цифр или символов), а также указать, как эти данные связаны между собой. В некоторых случаях можно также задать форматы и критерии проверки данных;
- ◆ **обработка данных** (Data manipulation) – данные можно обрабатывать самыми различными способами. Можно объединять данные с другой связанной с ними информацией и вычислять итоговые значения;
- ◆ **управление данными** (Data control) – можно указать, кому разрешено знакомиться с данными, корректировать их или добавлять новую информацию. Можно также определить правила коллективного пользования данными.

◆ Лабораторная работа № 1 Создание таблиц

Цель работы:

- получить понятие об объектах в MS-ACCESS;
- изучить структуру объекта MS-ACCESS “таблица”;
- научиться задавать полям различные типы данных.

Введение

Структура СУБД MS-ACCESS включает следующие объекты:

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

Таблицы состоят из полей и записей. Полями называются столбцы, а записями строки. Внести запись в таблицу означает заполнить данными какую-нибудь строку. Чтобы создать таблицу, необходимо определить ее поля, типы данных этих полей и, иногда, некоторые дополнительные свойства этих полей. Не все данные занимают в компьютере одинаковое место. Для их компактного хранения необходимо четко определить, что это: текст или число, дата или логический символ. В базах данных под каждый тип данных резервируется некоторое пространство и, если известно заранее, что оно не будет использовано до конца, его необходимо уменьшить. Как это сделать, Вы увидите по ходу выполнения данной работы.

Ход работы

1. Запустите программу MS ACCESS.
2. В области задач появившегося окна выберите команду «**Создать файл...**», затем - «**Новая база данных...**» (тоже самое можно сделать через меню «**Файл**»).
3. В следующем диалоговом окне Вы должны выбрать папку для базы данных и задать ее имя (например, «**Европа**»).
На экране появится окно с шестью вкладками, это и есть ваша база данных (она пока пустая).
4. Перейдите на вкладку «**Таблицы**».
5. Щелкните мышью по кнопке «**Создать**»

6. В появившемся диалоговом окне, выберите режим создания «Конструктор» и подтвердите данную операцию.

В столбец «имя поля» заносятся имена столбцов будущей таблицы (при этом нельзя использовать некоторые символы, в том числе точки и запятые). В столбце «тип данных» выбирается (с использованием кнопки вызова списка) тип данных. А то, что заносится в столбец «описание» затем появляется, в виде комментариев, в строке состояния (для проверки в одной из строк этого столбца напишите фразу: моя первая база данных).

Необходимо создать следующие поля:

| Имя поля | Тип данных |
|------------|------------|
| Код страны | Счетчик |
| Страна | Текстовый |
| Столица | Текстовый |
| Площадь | Числовой |
| Население | Числовой |
| Религия | Текстовый |
| Деньги | Текстовый |
| Строй | Текстовый |

Данные для ввода:

Албания

Площадь 28 748 кв. км
Столица Тирана
Число жителей, 3149000
Основная религия, Атеизм
Историческая ден. единица Лек
Гос. строй Республика

Андорра

Площадь 468 кв. км
Столица Андорра-ла-Вьеха
Число жителей 51 400
Основная религия Христианство (католики)
Ист. ден. единица Франк
Гос. строй Княжество

Бельгия

Площадь 30 518 кв. км
Столица Брюссель
Число жителей 9 865 000
Основная религия Христианство (католики)
Ист. ден. единица Франк
Гос. строй Монархия

Болгария

Площадь 110 994 кв. км
Столица София
Число жителей 8 978 000
Основная религия Христианство (православные)
Историческая ден. единица Лев
Гос. строй Республика

Дания

Площадь 43 092 кв. км
Столица Копенгаген
Число жителей 5 130 000
Основная религия Христианство (протестанты)
Ист. ден. единица Крона
Гос. строй Монархия

Финляндия

Площадь 338 145 кв. км
Столица Хельсинки
Число жителей 4 952 000
Основная религия Христианство (протестанты)
Ист. ден. единица Марка
Гос. строй Республика

Франция

Площадь 543 965 кв. км
Столица Париж
Число жителей 55 860 000
Основная религия Христианство (католики)
Ист. денежная единица Франк
Гос. строй Республика

Греция

Площадь 131 957 кв. км
Столица Афины
Число жителей 10 055 000
Основная религия Христианство (православные)
Ист. ден. единица Драхма
Гос. строй Республика

Италия

Площадь 301 277 кв. км
Столица Рим
Число жителей 57 401 000
Основная религия Христианство (католики)
Ист. денежная единица Лира
Гос. строй Республика

Югославия

Площадь 255 804 кв. км
Столица Белград
Число жителей 23 591 000
Основная религия Христианство (православные)
Ист. денежная единица Динар
Гос. строй Республика

Лихтенштейн

Площадь 160 кв. км
Столица Вадуц
Число жителей 27 840
Основная религия Христианство (католики)
Ист. ден. единица Франк
Гос. строй Монархия

Люксембург

Площадь 2 586 кв. км
Столица Люксембург
Число жителей 372 000
Основная религия Христианство (католики)
Ист. денежная единица Франк
Гос. строй Герцогство

Мальта

Площадь 316 кв. км
Столица Валлетта
Число жителей 347 000
Основная религия Христианство (католики)
Историческая ден. Фунт
Гос. строй Республика

Монако

Площадь 2 кв. км
Столица Монако
Число жителей 28 000
Основная религия Христианство (католики)
Ист. ден. единица Франк
Гос. строй Княжество

Нидерланды

Площадь 41 863 кв. км

Столица Амстердам

Число жителей 14 741 000

Основная религия Христианство
(католики)

Ист. денежная единица Флорин

Гос. строй Монархия

Норвегия

Площадь 323 878 кв. км

Столица Осло

Число жителей 4 202 000

Основная религия Христианство
(протестанты)

Ист. денежная единица Крона

Гос. строй Монархия

Польша

Площадь 312 683 кв. км

Столица Варшава

Число жителей 37 864 000

Основная религия Христианство
(католики)

Ист. денежная единица Злотый

Гос. строй Республика

Португалия

Площадь 92 389 кв. км

Столица Лиссабон

Число жителей 10 349 000

Основная религия Христианство
(католики)

Ист. денежная единица Эスクудо

Гос. строй Республика

Румыния

Площадь 237 500 кв. км

Столица Бухарест

Число жителей 23 014 000

Основная религия Христианство
(православные)

Историческая ден. единица Лей

Гос. строй Республика

Сан-Марино

Площадь 61 кв. км

Столица Сан-Марино

Число жителей 22 830

Основная религия Христианство
(католики)

Историческая ден. единица Лира

Гос. строй Республика

Швейцария

Площадь 41 293 кв. км

Столица Берн

Число жителей 6 626 000

Основная религия Христианство
(католики)

Ист. денежная единица Франк

Гос. строй Республика

Испания

Площадь 504 783 кв. км

Столица Мадрид

Число жителей 38 996 000

Основная религия Христианство
(католики)

Ист. денежная единица Песета

Гос. строй Монархия

Великобритания

Площадь 244 110 кв. км

Столица Лондон

Число жителей 57 006 000

Основная религия Христианство
(протестанты)

Историческая ден. единица Фунт

Гос. строй Монархия

Швеция

Площадь 449 964 кв. км

Столица Стокгольм

Число жителей 8 415 000

Основная религия Христианство
(протестанты)

Ист. денежная единица Крона

Гос. строй Монархия

Чехословакия

Площадь 127 900 кв. км

Столица Прага

Число жителей 15 604 000

Основная религия Христианство
(католики)

Ист. денежная единица Крона

Гос. строй Республика

Венгрия

Площадь 93 031 кв. км

Столица Будапешт

Число жителей 10 591 000

Основная религия Христианство
(католики)

Ист. денежная единица Форинт

Гос. строй Республика

Германия

Площадь 357 042 кв. км

Столица Бонн

Число жителей 77 370 000

Основная религия Христианство
(протестанты)

Ист. денежная единица Марка

Гос. строй Республика

Австрия

Площадь 83 857 кв. км

Столица Вена

Число жителей 7 557 000

Основная религия Христианство
(католики)

Ист. денежная единица Шиллинг

Гос. строй Республика

7. После ввода имен полей и указания типов хранимых в них данных желательно задать ключевое поле. Так как значения в ключевом поле должны быть уникальными, т.е. не повторяющимися для отдельных записей. В этом качестве следует выбрать поле **код страны**.

Для задания ключевого поля необходимо щелкнуть правой клавишей мыши по полю **код страны** и в появившемся меню выполнить команду **«Ключевое поле»**.

8. Закройте окно конструктора и, при запросе о сохранении, задайте имя таблицы - Страны Европы.
9. Откройте таблицу для заполнения данными. (Заметьте, что поле с типом данных счетчик будет заполняться автоматически).
10. Посчитайте максимальное количество символов в каждом из полей с текстовыми данными и, на всякий случай, прибавьте к полученным значениям 5.
11. Откройте таблицу в режиме конструктора.
12. По очереди, переводя текстовый курсор в каждое из текстовых полей, задайте в окне свойств поля его размер (значения, полученные в пункте 10).
13. Просмотрите, какие еще настройки можно задавать для текстовых полей.
14. Просмотрите, какие настройки можно задавать для полей других типов.
15. Закройте таблицу с сохранением.
16. Сохраните данную базу данных на диске для следующей лабораторной работы. В отличие от ранее изученных программ, для этого необходимо закрыть ACCESS и скопировать файл базы одним из средств Windows.

Контрольные вопросы

1. Что называется базой данных?
2. Чем поле отличается от записи?
3. Что такое СУБД и каковы ее основные функции?
4. Какие объекты ACCESS Вы знаете?
5. Для чего служит объект "таблица"?
6. Для чего данным задаются различные типы?
7. Что можно настроить в свойствах поля таблицы?

Лабораторная работа № 2

Реляционные базы данных

Цель работы:

- получить понятие о реляционных базах данных и основных возможностях, предоставляемых ими.

Введение

В процессе ввода данных очень часто возникает необходимость защищить оператора от ошибки и разграничить доступ к важной информации. Для этого данные разных категорий разделяют по разным таблицам, кроме того, как правило, такое разделение позволяет более компактно хранить информацию. Данные в главной таблице индексируют и, используя эти индексы, связывают таблицу с подчиненными таблицами. Базы данных с такими связями называют реляционными. В данной лабораторной работе созданная ранее база данных будет приводится к такому виду.

Ход работы

Разделение данных на две таблицы

1. Откройте базу данных **Европа**.
2. Создайте таблицу **Религия** с полями:

| Имя поля | Тип данных |
|-----------------|-------------------|
| Код_религии | Счетчик |
| Религия | Текстовый |

3. Поле **Код_религии** сделайте ключевым
4. Создайте таблицу **Строй** с полями:

| Имя поля | Тип данных |
|-----------------|-------------------|
| Код_строя | Счетчик |
| Строй | Текстовый |

5. Поле **Код_строя** сделайте ключевым.
6. Заполните эти таблицы (напротив названия каждой религии и каждого строя будет стоять код).
7. Откройте таблицу **Страны Европы** и замените названия в полях **Строй** и **Религия** на соответствующие им коды в ранее созданных таблицах (для автоматизации попробуйте использовать команду замены из меню «Правка»).
8. Откройте таблицу **Страны Европы** в режиме конструктора.
9. Для полей **Строй** и **Религия** измените тип данных на числовой, одновременно установив размер поля **Длинное целое** (так как связываемые поля, как правило, должны иметь одинаковый тип данных).

Установка связи между двумя таблицами

1. Выполните команду «**Схема данных**» из меню «**Сервис**».
2. С помощью диалогового окна «**Добавление таблицы**» добавьте в схему все три таблицы.
3. На поле **Код_религии** таблицы **Религия** нажмите левую клавишу мыши и, удерживая ее, перетащите на поле **Религия** таблицы **Страны Европы**.

В появившемся диалоговом окне необходимо установить нужную связь: флажок напротив опции **обеспечение целостности данных** означает, что перед тем, как занести данные в подчиненную таблицу, программа будет проверять их на соответствие главной. (Таблица **Страны Европы** является подчиненной для таблиц **Религия** и **Строй**). Флажок напротив опции **каскадное обновление связанных полей** означает, что изменения в главной таблице автоматически будут влиять на подчиненную. Флажок напротив опции **каскадное удаление связанных полей** означает, что поля, удаленные в главной таблице, будут удалены и в подчиненной.

4. Установите все эти флагки.
5. Аналогичную операцию проделайте с таблицами **Страны Европы** и **Строй** (Если связь не устанавливается, еще раз проверьте типы данных связываемых полей).
6. Закройте схему данных с сохранением.
7. Попробуйте в таблице **Страны Европы** в полях **Строй** и **Религия** поменять коды на несуществующие в главных таблицах. Получилось? Почему?
8. Сохраните базу данных на дискету.

Контрольные вопросы

1. Для чего данные разбивают на несколько таблиц?
2. Какое необходимое условие существует для создания связи с обеспечением целостности данных?

3. Каким образом при разбиении достигается компактность хранения информации?

Лабораторная работа № 3

Создание запросов на выборку данных

Цель работы:

- изучить объекты баз данных **Запрос**;
- научиться пользоваться простейшими логическими операндами;
- получить практику выборки заданной информации из таблиц.

Введение

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

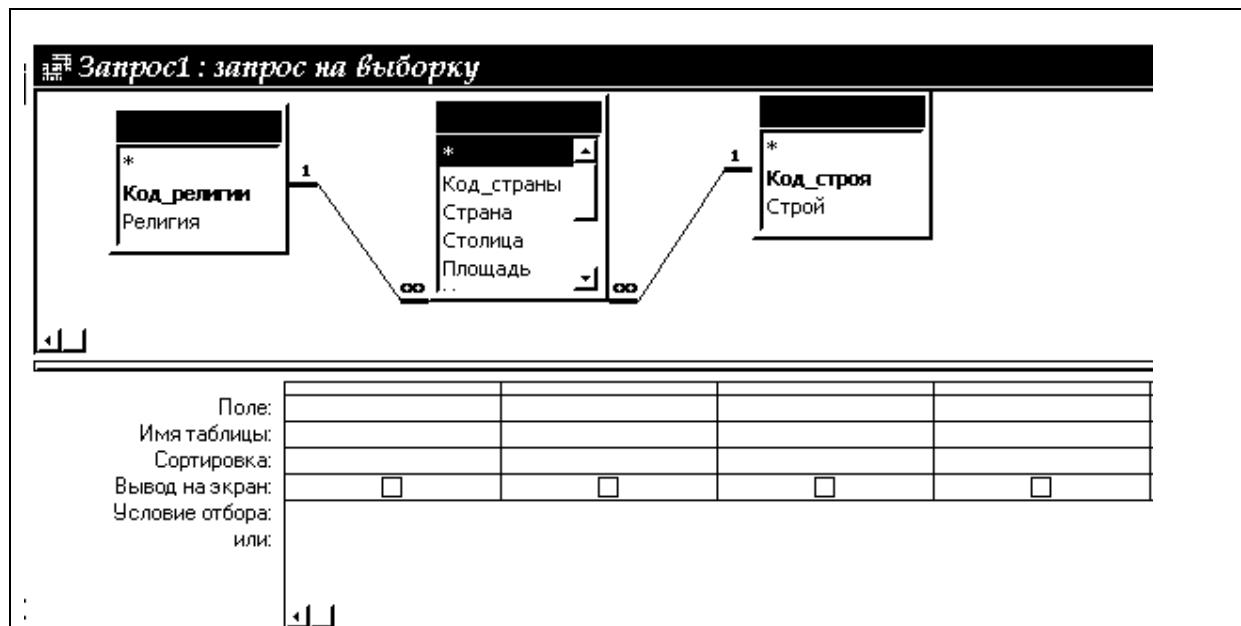
В **запросах на выборку** данные могут: отбираться по многим критериям; сортироваться; с ними могут производиться вычислительные операции.

Запрос это времененная таблица. Это значит, что данные в них не хранятся постоянно, а только временно вызываются из таблиц, по заранее заданному шаблону, в момент активизации запроса. Таким образом, в базе данных постоянно хранится только шаблоны вызова данных (временные таблицы удаляются после закрытия запроса), а сама информация не дублируется.

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

Ход работы

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



Теперь необходимо выбрать поля таблиц, включаемые в запрос. Из таблицы **Страны Европы** это поля: **Страна, Столица, Площадь, Население, Деньги**. Из таблицы **Строй** поле **Строй**, а из таблицы **Религия** поле **Религия**, так как в этих таблицах нужные данные хранятся в виде текста.

6. Первое поле добавьте, произведя двойной щелчок по его имени в таблице.
7. Второе перетащите, используя мышь (есть и третий способ, но он менее удобен).
8. Остальные поля добавьте любым способом.
9. Закройте запрос, при закрытии задав ему имя **Основной**.
10. Откройте запрос и просмотрите выводимые данные.
11. Закройте запрос.

Теперь необходимо настроить запрос на различные условия выборки. Обычно для этого используют один запрос, меняя эти условия. Но для проверки выполненной работы следует создать несколько запросов (на каждое условие по запросу). Чтобы не создавать заново почти одинаковые запросы, удобно несколько раз скопировать **Основной** через буфер обмена.

Для того, чтобы выбрать из таблицы **Страна Европы** только государства с определенной религией, необходимо для поля запроса **Религия** в строке **Условие отбора** ввести название этой религии. Аналогично для любого поля.

Для того, чтобы запретить вывод в запросе информации о странах с определенной религией, необходимо для данного поля в уже указанной строке ввести название заданной религии, но перед ним поставить оператор **Not**.

Для того, чтобы вывести информацию о странах с населением больше 1000000, достаточно для поля **Население** в строке **Условие отбора** ввести >1000000. Кстати, для условий предусмотрено две строки, поэтому для одного поля можно вводить два условия.

Можно сделать так, чтобы перед выводом запроса программа просила ввести данные, записи с которыми будут выведены. Для этого для нужного поля в строке **условие отбора** надо ввести: [Введите данные для отбора]. Выражение в скобках может быть любым, но именно оно появится в диалоговом окне.

Для того, чтобы находить в поле не конкретное значение, а только его фрагмент, используют оператор **Like**. Его ставят впереди заданного фрагмента, а до или после фрагмента можно использовать символы маски (например, символ ? означает один произвольный символ, символ * - любое число произвольных символов).

12. Выделите запрос **Основной** (он должен быть закрыт).
13. Скопируйте его в буфер обмена.
14. Вставьте его в базу 6 раз, задавая имена, созвучные следующим ниже заданиям.
15. Создайте запрос, выводящий страны с **Православием**.
16. Создайте запрос, не выводящий страны с **Православием**.
17. Создайте запрос, выводящий страны с населением более **10000000** и менее **200000000**.
18. Создайте запрос, выводящий страны с населением более **3000000** и площадью менее **30000** кв. км.
19. Создайте запрос, спрашивающий: страны с какой денежной единицей вывести на экран?
20. Создайте запрос, выводящий страны с денежной единицей, название которой содержит букву «к».
21. Сохраните базу данных на дискету.

Контрольные вопросы

1. Для чего служат запросы на выборку?
2. Почему запросы не занимают в базе данных много места?
3. По какому количеству критериев можно отбирать информацию для одного поля запроса?
4. Для чего в запрос вставляют ссылки на таблицы?
5. Какие способы вставки полей в запрос вы знаете?
6. Для чего служит оператор **Like**?

Лабораторная работа № 4

Создание запросов с вычисляемыми полями

Цель работы:

- практика создания баз данных;
- изучение построения математического выражения в запросах;
- изучение использования групповых операций в запросах;
- практика работы с контекстными меню.

Введение

Как известно, базы данных нужны не только для хранения, но и для обработки данных. Одним из способов обработки данных является создание вычисляемых полей в запросах на выборку. Этого способа иногда бывает достаточно для создания базы данных, имеющей практическое значение, например, с помощью запросов рассчитываются и заносятся в базу данные о продаже товаров и прибыли маленького магазина. В данной работе опущены многие подсказки по предыдущим темам. Вам самим предстоит решать, какие поля делать ключевыми и т.п.

Ход работы

Часть 1 (подготовка).

1. Создайте базу данных **Магазин**.
2. Создайте таблицу **Товар** по приведенному ниже примеру.

| код товара | товар | цена поставки \$ |
|------------|------------|------------------|
| 1 | монитор | 150,00 |
| 2 | винчестер | 120,00 |
| 3 | CD-ROM | 70,00 |
| 4 | Клавиатура | 11,00 |
| счетчик | текстовый | числовой |

← типы данных

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

Примечание 1: после задания типа данных для поля **наценка** переведите текстовый курсор в поле **Размер поля** и с помощью **кнопки выпадающего списка** выберите режим **«С плавающей точкой (4 байт)»**.

Примечание 2: после задания типа данных для поля **дата** переведите текстовый курсор в поле **Формат поля** и с помощью **кнопки выпадающего списка** выберите режим **Краткий формат даты**.

| код товара | количество | наценка | дата |
|------------|------------|----------|------------|
| 1 | 5 | 0,2 | 04.10.2006 |
| 1 | 5 | 0,2 | 05.10.2006 |
| 2 | 2 | 0,2 | 04.10.2006 |
| 3 | 10 | 0,2 | 04.10.2006 |
| 4 | 2 | 0,02 | 02.10.2006 |
| числовой | числовой | числовой | дата/время |

← типы данных

4. Создайте таблицу **Курс доллара** по приведенному ниже примеру.

| дата | курс \$ |
|------------|---------|
| 04.10.2006 | 25р. |
| 05.10.2006 | 26р. |
| 06.10.2006 | 27р. |
| 02.10.2006 | 24р. |
| 03.10.2006 | 0р. |

← типы данных

5. Используя команду **Схема данных** из меню **Сервис** свяжите таблицы **Товар и Продажа товара** дате по полю «**код товара**».
6. Создайте в режиме **Конструктора** запрос **прибыль по дате за вид товара** с полями **товар** из таблицы **Товар** и **дата** из таблицы **Курс доллара**.

ВНИМАНИЕ: Убедитесь, что в запрос вставлены сами эти таблицы.

Кроме того, для дальнейшей работы необходимо, чтобы в этот запрос была вставлена таблица **Продажа товара по дате**.

7. Установите (перетаскиванием мыши) связь между полями **Дата** таблиц **Продажа товара по дате** и **Курс доллара**.
8. Сохраните запрос под указанным именем.

Часть 2 (построение вычисляемого поля)

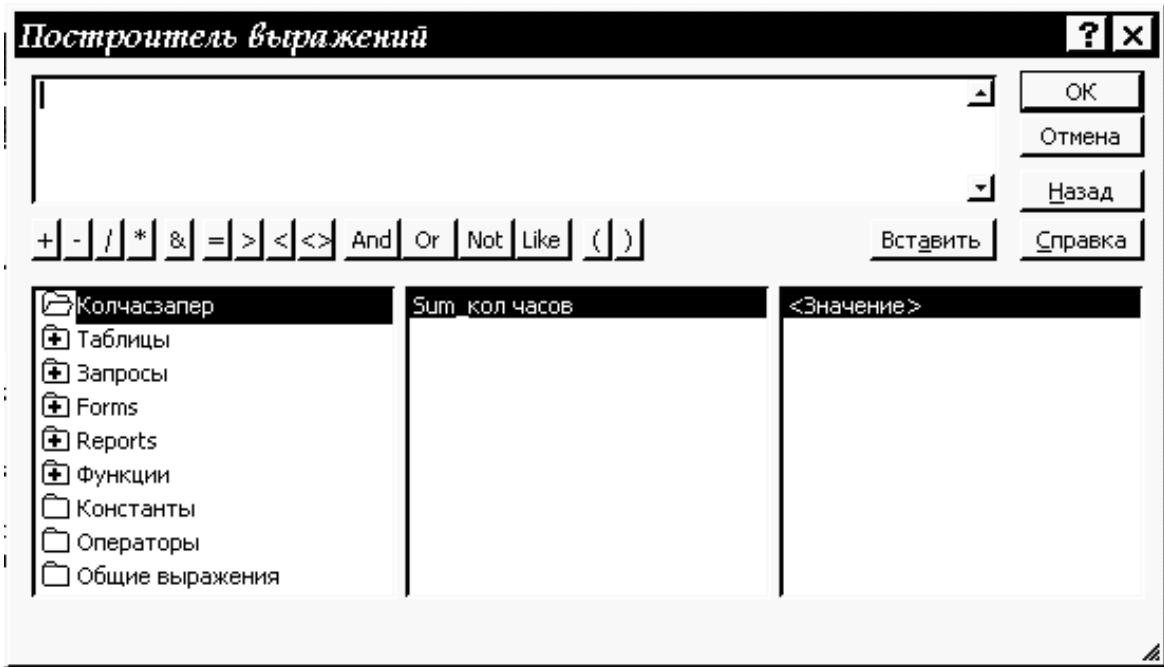
Необходимо (как видно из названия запроса) построить в одном из столбцов данного запроса выражение для расчета прибыли по отдельному виду товара за определенный день в рублях. Для этого необходимо умножить данные столбца цен в долларах на курс доллара по датам и умножить полученное произведение сначала на столбец коэффициентов наценки, затем - на количество проданного за этот день товара.

Решение задачи:

1. Откройте запрос **Прибыль по дате за вид товара** в режиме конструктора.

2. Установите текстовый курсор в строку **Поле** свободного столбца и щелчком правой клавиши мыши на нем вызовите контекстное меню.
3. Выполните команду **Построить**.

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



4. В данном диалоговом окне, открывая папки **Таблицы** и **Запросы**, а затем - заданные таблицы и запросы, необходимо, используя кнопку «**Вставить**», добавлять в выражение нужные поля (из этих таблиц и строк), а также знаки необходимых математических операций.
5. Подготовив выражение, как указано в описании ко второй части, нажать подтверждающую клавишу.
6. Установить флажок (щелчком мыши) в строке **Вывод на экран** создаваемого столбца.
7. Сохраните измененный запрос (**Файл – сохранить**).
8. Переведите текстовый курсор в начало строки **Поле** созданного столбца.
9. Удалите название поля (Выражение1), сохранив двоеточие.
10. Введите название поля: **прибыль по дате за вид товара**
11. В меню **Вид** выполните команду **Режим таблицы**.
12. Если построенное поле выводит данные, закройте запрос с сохранением, иначе закройте без сохранения.

Рассчитывается прибыль магазина по виду товара за определенную дату. Теперь необходимо рассчитать полную прибыль магазина по дате. Для этого применяется функция группировки в создаваемом запросе. В такой запрос обычно вставляют поле (или поля), по которым будет производиться группировка и поле, значения которого являются группировочным при-

знаком. В нашем случае группировка осуществляется по полю **Дата**, а групповая операция (суммы) применяется к полю, построенному в предыдущем запросе.

Решение задачи:

1. Создать запрос, добавив в него только предыдущий запрос (из него будут браться данные).
2. Выбрать поле **Дата** и поле, построенное в предыдущем запросе.
3. Выбрать на панели инструментов или в контекстном меню команду **Групповые операции** (в запросе должна добавиться строка **Групповая операция**).
4. Установить текстовый курсор в ячейку на пересечении этой строки и столбца с выражением, построенным в предыдущем запросе.
5. Щелкнуть в ней по кнопке выпадающего списка и выбрать из него функцию **sum**.
6. Закрыть запрос с сохранением.
7. Сохраните базу данных на дискету.

Контрольные вопросы

1. Как вызывается контекстное меню?
2. Для чего предназначена операция группировки?
3. Что значит связать поля с сохранением целостности данных?
4. Можно ли установить такую связь между полями с разными типами данных; и если нет, то почему?

Лабораторная работа № 5
Создание и конструирование форм

Цель работы:

- знакомство с объектами **форма**;
- практика создания форм;
- практика конструирования форм;
- знакомство с режимом **мастера создания форм**.

Введение

Форма это структура базы данных, позволяющая удобно вводить и обрабатывать заданную информацию. Очень часто для компактного хранения информации требуется заносить в таблицу код объекта, а не его имя. Таким примером может служить таблица **Продажа товара по дате** из предыдущей лабораторной работы. Заполнение таких таблиц, особенно для рядовых операторов, было бы связано с большими проблемами. Формы

же позволяют, выбирая из списка имена объектов, вставлять в таблицы их коды. Возможности форм необъятны для данного практикума, в нем рассматриваются лишь некоторые из них. Кроме того, в этой работе используются некоторые элементы визуального программирования: берутся готовые элементы управления (поля, поля со списками, кнопки и т.п.), затем им задаются определенные свойства, а также связи с данными, находящимися в таблицах и запросах. Данная тема будет продолжена и в следующих работах.

Ход работы

Часть 1 (подготовка)

1. Откройте базу данных **Магазин**.
2. Убедитесь в наличии следующих таблиц.

Товар

| код товара | товар | цена поставки \$ |
|------------|------------|------------------|
| 1 | монитор | 150,00 |
| 2 | винчестер | 120,00 |
| 3 | CD-ROM | 70,00 |
| 4 | клавиатура | 11,00 |

счетчик текстовый

числовой

← типы данных

Продажа товара по дате

| код товара | количество | наценка | дата |
|------------|------------|---------|------------|
| 1 | 5 | 0,2 | 04.10.2006 |
| 1 | 5 | 0,2 | 05.10.2006 |
| 2 | 2 | 0,2 | 04.10.2006 |
| 3 | 10 | 0,2 | 04.10.2006 |
| 4 | 2 | 0,02 | 02.10.2006 |

числовой

числовой

числовой

дата/время

← типы данных

Курс доллара

| дата | курс \$ |
|------------|---------|
| 04.10.2006 | 25р. |
| 05.10.2006 | 26р. |
| 06.10.2006 | 27р. |
| 02.10.2006 | 24р. |

| | | |
|------------|----------|--------------------|
| дата/время | денежный | ← типы данных |
|------------|----------|--------------------|

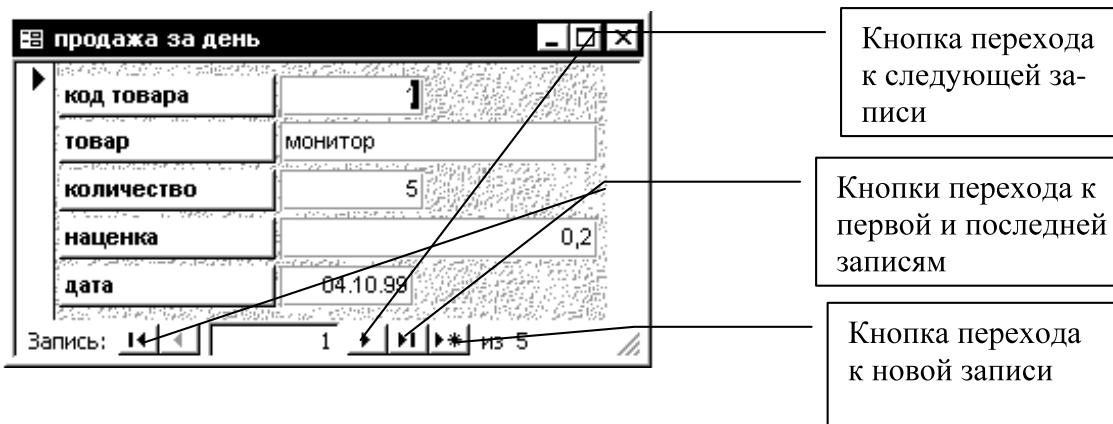
3. Проверьте (в схеме данных) наличие связи между полями **код товара** в таблицах **Товар** и **Продажа товара по дате**.

Часть 2 (создание формы).

Формы и отчеты довольно сложные структуры баз данных, поэтому обычно их создают в режиме **Мастера** и дорабатывают в режиме **Конструктора**.

1. Перейдите на вкладку **Формы**.
2. Щелкните по кнопке **Создать** и выберите режим **Мастер форм**.
3. В поле со списком **Таблицы/запросы** выберите таблицу – источник: **Продажа товара по дате**.
4. Используя клавишу «>>», перенесите все поля данной таблицы из окна **доступные поля** в окно **выбранные поля** и перейдите к следующему диалоговому окну.
5. В нем установите флажок напротив опции **в один столбец** и перейдите к следующему диалоговому окну.
6. Выберите в нем оформление и перейдите к следующему диалоговому окну.
7. В последнем окне задайте имя форме и нажмите на кнопку **Готово**.

Вы создали форму, которая позволяет вводить данные в таблицу **Продажа товара по дате**.



Но для ее заполнения необходимо помнить коды всех товаров. Поэтому следующая задача - обеспечить удобный ввод данных в поле **код товара** таблицы **Продажа товара по дате**.

Часть 3 (доработка формы).

Необходимо обеспечить, чтобы при вводе названия товара в форме в таблицу заносился его код.

1. Откройте вашу форму в режиме **конструктора**.
2. Выделите поле **код товара** и удалите его, используя клавишу **Delete**.
3. Найдите в окне программы панель элементов и выпишите названия объектов, доступных для использования в формах.

4. Выберете **поле со списком**.
5. Подведите курсор мыши к предполагаемому месту вставки нового поля (учтите, что слева от него автоматически вставится заголовок).
6. Нажмите левую клавишу мыши, и удерживая ее, растяните объект как рисунок (после того, как вы отпустите ее, у вас появится первое окно диалога).
7. Ознакомьтесь с ним и установите флажок напротив нужной опции (верхняя).
8. Нажав клавишу **Далее**, перейдите к следующему диалоговому окну.
9. В нем выберите таблицу, откуда будет браться наименование товара.
10. В следующем диалоговом окне выберите два поля (из одного будут браться коды товаров, из другого – их названия, которые и будут отображаться в форме, т.е. поля **код товара и товар**).
11. Со следующим диалоговым окном просто ознакомьтесь.
12. В следующем выберите опцию **Сохранить в поле** и, используя кнопку вызова списка, выберите из него **код товара**.
13. В последнем окне задайте подпись созданного поля (**товар**).
14. Закройте форму с сохранением.
15. Откройте ее и произведите ввод новой записи.
16. Проверьте, что занеслось в таблицу **Продажа товара по дате** в поле **код товара**.
17. Скопируйте базу данных на дискету.

Контрольные вопросы

1. Для чего служит объект **Форма**?
2. Какие элементы можно вставлять в форму в режиме конструктора?
3. Как в форме осуществляется переход между записями?
4. Подумайте, какие недостатки есть у созданной Вами формы?

Лабораторная работа № 6

Свойства форм

Цель работы:

- знакомство с **формами**;
- практика создания форм;
- практика конструирования форм;
- знакомство с режимом **мастера создания форм**.

Введение

В предыдущей лабораторной работе один из контрольных вопросов звучал так: «Подумайте, какие недостатки есть у созданной Вами формы?»; ответ прост:

- при добавлении новых записей можно испортить предыдущие;
- порядок ввода полей записи нарушен созданием нового поля;
- шрифт мелок и невыразителен.

Для большинства объектов MS Windows и программ этой операционной системы предусмотрена возможность задания неких параметров (свойств). Самый простой способ для их изменения - вызвать диалоговое окно щелчком правой клавиши мыши на заданном объекте (контекстное меню).

Ход работы

1. Сделайте две копии своей формы. Для этого:
 - 1.1 в меню **правка** выберите команду **копировать**;
 - 1.2 в меню **правка** выберите команду **вставить**;
 - 1.3 в появившемся диалоговом окне задайте имя **ввод новых данных**;
 - 1.4 повторите команду **вставить** и задайте имя **удаление записей**;
 - 1.5 своей первой форме задайте имя **редактирование данных**.
2. Откройте форму **ввод новых данных** в режиме конструктора.
3. Щелкните правой клавишей мыши на заголовке окна формы.
4. Выберите из контекстного меню команду **свойства**.
5. В появившемся диалоговом окне перейдите на вкладку **все**.
6. Переведите текстовый курсор в строку **разрешить изменение** и, используя кнопку вызова списка, выберите **нет**.
7. Аналогичную операцию проделайте в строке **разрешить удаление**.
8. Закройте форму с сохранением, откройте ее и попробуйте:
 - изменить запись;
 - удалить запись;
 - добавить запись.
9. Подумайте: какие изменения надо внести в две другие формы. Сделайте их.
10. Откройте одну из форм в режиме конструктора.
11. Вызовите на заголовке ее окна контекстное меню.
12. Выполните команду **последовательность перехода**.
13. В появившемся диалоговом окне измените последовательность так, чтобы поля шли в порядке расположения внутри формы.
14. Проверьте сделанные изменения.
15. Откройте любую форму в режиме конструктора.
16. Вызовите контекстное меню на одном из полей формы.
17. Выполните команду **свойства** и в появившемся окне попробуйте самостоятельно изменить величину шрифта.

Контрольные вопросы

1. Как вызвать диалоговое окно свойств объекта?
2. Как вы думаете: какие команды содержит контекстное меню?

3. Какие команды вы использовали при размножении форм?
4. Что вы еще смогли изменить в форме?

Лабораторная работа № 7

Создание кнопочных форм

Цель работы:

- получить практику по созданию форм;
- получить представление о создании интерфейсных оболочек для баз данных.

Введение

После создания реальной базы данных грамотно пользоваться ею может только создатель. Поэтому для ее эксплуатации рядовым пользователем необходимо создать оболочку управления с понятным интерфейсом (интерфейс – средства взаимодействия). Для этой цели подходят так называемые кнопочные формы. Идея в том, чтобы установить связь между определенным событием и щелчком мыши по графическому изображению кнопки. Однако не все события можно вызвать с использованием кнопочных форм, реализованных в СУБД ACCESS. Для обхода этого препятствия можно воспользоваться макросами. Макрос является программой, содержащей макрокоманды, т.е. вызов определенных действий с заданными параметрами. Запуск макроса можно выполнить из кнопочной формы.

Часть 1 (подготовка).

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

Часть 2 (создание кнопочной формы).

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

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

Часть 3 (Создание макроса, открывающего запрос).

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

Часть 4 (создание кнопочной команды, запускающей макрос).

1. Выполните команду **диспетчер кнопочных форм** из списка **надстроек** меню **сервис**.
2. В появившемся диалоговом окне нажмите кнопку **изменить**.
3. В следующем окне выполните команду **создать**.
4. Выполните настройки кнопочной формы аналогично части 2.

Часть 5 (изменение параметров запуска).

Теперь необходимо сделать недоступным при открытии файла окно базы данных, обеспечив, при этом, автоматический запуск кнопочной формы. Для этого необходимо проделать следующие действия:

1. Выполнить команду **параметры запуска** из меню **сервис**.
2. В появившемся диалоговом окне убрать флажок напротив **окно базы данных**.
3. А в **форма** выставить кнопочную форму.

Контрольные вопросы

1. Для чего служат кнопочные формы?
2. Можно ли создать в базе две кнопочные формы?
3. Для чего создавался макрос?
4. Какие события можно задавать по нажатию кнопки в ACCESS?

ЛИТЕРАТУРА

1. Острайковский В.А. Информатика. Учебное пособие для вузов. - М.: Высшая школа, 2001. 501с.
2. Информатика: Базовый курс / С.В. Симонович и др. – СПб.: Питер, 2002. – 640 с.
3. Информатика. / А.Н. Степанов. – СПб.: Питер, 2002. – 608 с.
4. Лисяк Н.К., Лисяк В.В. Основы информатики. Учебное пособие. Таганрог: Изд-во ТРТУ, - 2003. 122с.
5. Акулов О.А., Медведев Н.В. Информатика. Базовый курс. М.:Омега-Л, 2004. -552с.
6. Microsoft Office Specialist – Учебный курс Office 2003. Практ. пособие., пер. с англ. – М.: ЭКОМ;БИНОМ. Лаборатория знаний, 2006. – 1006 с.: илл.
7. Моркес Д. Microsoft Access 2003. Эффективный самоучитель / Пер. с чеш. Серия «Просто о сложном». – СПб.: Наука и техника, 2006. – 352 с., ил.

Приложение 1

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

САНКТ – ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ

Кафедра - «Вычислительной техники и информационного
обеспечения АПК»
(ВТ и ИО АПК)

Дисциплина: «ИНФОРМАТИКА»

ЛАБОРАТОРНАЯ РАБОТА №_____

Тема: _____

Выполнил(а): студент(ка). _____ группы

Фамилия И.О.

Проверил(а): _____

Санкт – Петербург
Пушкин (Царское село)
200_ г.

Порядок оформления выполнения лабораторной работы.

1. Оформить титульный лист.
2. Условие лабораторной работы. (Переписать из методической разработки).
3. Задание № ____ (условие)
 - 3.1. Переписать условие задания.
 - 3.2. Выполнить каждый пункт с описанием последовательности выполнения на ПЭВМ и включением в отчет рисунков - экранных форм решения с объяснением и пояснениями к рисункам, таблицам, схемам.
 - 3.3. Привести общее заключение по лабораторной работе с выводами и предложениями по данной теме.
4. Включить верхний и нижний колонтитулы. В верхнем колонтитуле указать № группы и ФИО студента, в нижнем колонтитуле - № лабораторной работы и тему.

СОДЕРЖАНИЕ

| | | |
|---|-------|----|
| ПРЕДИСЛОВИЕ | | 3 |
| ВВЕДЕНИЕ | | 4 |
| ЛАБОРАТОРНАЯ РАБОТА №1 СОЗДАНИЕ ТАБЛИЦ | | 5 |
| ЛАБОРАТОРНАЯ РАБОТА №2 РЕЛЯЦИОННЫЕ БАЗЫ ДАННЫХ | | 10 |
| ЛАБОРАТОРНАЯ РАБОТА №3 СОЗДАНИЕ ЗАПРОСОВ НА ВЫБОРКУ ДАННЫХ | | 12 |
| ЛАБОРАТОРНАЯ РАБОТА №4 СОЗДАНИЕ ЗАПРОСОВ С ВЫЧИСЛЯЕМЫМИ ПОЛЯМИ | | 15 |
| ЛАБОРАТОРНАЯ РАБОТА №5 СОЗДАНИЕ И КОНСТРУИРОВАНИЕ ФОРМ | | 18 |
| ЛАБОРАТОРНАЯ РАБОТА №6 СВОЙСТВА ФОРМ | | 21 |
| ЛАБОРАТОРНАЯ РАБОТА №7 СОЗДАНИЕ КНОПОЧНЫХ ФОРМ | | 23 |
| ЛИТЕРАТУРА | | 25 |
| ПРИЛОЖЕНИЕ 1 | | 26 |

This document was created with Win2PDF available at <http://www.daneprairie.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.