

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кузбасский государственный технический университет
имени Т. Ф. Горбачева»

Кафедра информационных
и автоматизированных производственных систем

Составители
Игнатьева Е. А. Ивина О. А.
Кулак И. В. Башкирцева Е. В.
Назимов А. С. Рябова Е. А. Асанов С. А.

ОСНОВЫ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Методические указания для практических занятий

Рекомендовано учебно-методической комиссией направления
09.03.02 «Информационные системы и технологии»,
в качестве электронного издания
для использования в учебном процессе

Кемерово 2023

Рецензенты

Чичерин И. В. – заведующий кафедрой информационных и автоматизированных производственных систем.

Фомин А. И. – профессор, доктор технических наук, председатель учебно-методической комиссии направления 20.03.01 «Техносферная безопасность»

Игнатьева Елена Александровна Ивина Оксана Анатольевна

Башкирцева Елена Владимировна Кулак Илья Валерьевич

Назимов Александр Сергеевич Рябова Елена Анатольевна

Асанов Сергей Александрович

Основы информационных технологий : методические указания для практических занятий : для обучающихся всех направлений всех форм обучения/ сост. Е. А. Игнатьева, О. А. Ивина, Е. В. Башкирцева, И. В. Кулак, А. С. Назимов, Е. А. Рябова, С.А. Асанов, Кузбасский государственный технический университет имени Т. Ф. Горбачева. – Кемерово, 2023. – Текст : электронный.

В данных методических указаниях изложено содержание практических работ, порядок их выполнения и контрольные вопросы к ним.

© Кузбасский государственный
технический университет
имени Т. Ф. Горбачева, 2023

© Игнатьева Е.А., Ивина О. А.,
Башкирцева Е.В., Кулак И.В.,
Назимов А.С., Рябова Е.А.,
Асанов С. А., составление, 2023

Практическое занятие №1

Основы позиционных систем счисления

Цель работы: изучить многообразие систем счисления и научиться переводить числа из одной системы счисления в другую.

Теоретическая часть

Система счисления – это способ записи чисел с помощью определенного набора символов. Обычно в качестве символов используются цифры, а совокупность цифр, необходимых для записи числа в той или иной системе счисления – **алфавитом системы счисления**.

Система счисления должна обладать следующими свойствами:

- простота способа записи чисел на материальном носителе;
- удобство выполнения арифметических операций над числами в предложенной записи;
- наглядность обучения основам работы с числами.

Классификация систем счисления

Все многообразие систем счисления можно разделить на две группы:

- Аддитивные (непозиционные) системы счисления.
- Позиционные системы счисления.

Аддитивные системы счисления – системы записи чисел, в которых каждой цифре в записи числа соответствует величина, не зависящая от местонахождения этой цифры в записи числа.

Наиболее известной из аддитивных систем счисления является римская система счисления. К аддитивным системам счисления относят также славянская система счисления, китайская система счисления, система счисления племени Майя и другие.

Позиционные системы счисления – системы записи чисел, в которых значение цифры в записи числа зависит от ее позиции или местонахождения в числе.

Совокупность различных цифр, используемых в позиционной системе счисления для записи чисел, называется *алфавитом системы счисления*.

Базис позиционной системы счисления – это последовательность чисел, каждое из которых задает значение цифры по ее месту в записи числа, то есть «вес» каждого разряда.

Приведем в качестве примера базисы некоторых традиционных систем счисления:

Десятичная система счисления: $1, 10, 10^2, 10^3, \dots, 10^n$.

Двоичная система счисления: $1, 2, 2^2, 2^3, \dots, 2^n$.

Восьмеричная система счисления: $1, 8, 8^2, 8^3, \dots, 8^n$.

Позиционные системы счисления бывают традиционными и нетрадиционными.

Системы счисления, базис которых образуют члены геометрической прогрессии, называют традиционными.

К традиционным системам счисления относятся двоичная, восьмеричная, десятичная и шестнадцатеричная системы счисления. Для записи чисел в десятичной системе счисления используются цифры 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. В восьмеричной системе счисления: 0, 1, 2, 3, 4, 5, 6, 7, а восьмерка записывается двумя цифрами 10. Для записи чисел в шестнадцатеричной системе счисления необходимо располагать уже 16-ю различными символами, используемыми как цифры. В качестве первых десяти шестнадцатеричных цифр используются те же, что и в десятичной системе (0, 1, ..., 9), для обозначения остальных шести цифр (в десятичной системе они соответствуют числам 10, 11, 12, 13, 14, 15) используют буквы латинского алфавита – A, B, C, D, E, F.

К нетрадиционным системам счисления относятся факториальная, фибоначчиевая и уравновешенная. Базисы этих систем счисления выглядят следующим образом:

- ✓ факториальная – $1!, 2!, \dots, (n-1)!, n!$;
- ✓ фибоначчиевая – $1, 2, 3, 5, 8, 13, 21, \dots$;
- ✓ уравновешенная троичная – $-1, 0, 1$.

Преобразование чисел

В общем случае перевод любого числа, состоящего из целой и дробной частей

$$A_P = \pm(a_n P^n + a_{n-1} P^{n-1} + \dots + a_1 P^1 + a_0 P^0 + a_{-1} P^{-1} + \dots + a_{-m} P^{-m}), \quad (1)$$

из системы счисления с основанием P_1 в систему с основанием P_2 может быть выполнен по универсальному алгоритму, который нетрудно определить, приведя целую и дробную части выражения к видам:

$$A_P^y = \pm(((\dots((a_n P + a_{n-1})P + a_{n-2})P + \dots)P + a_0) \quad (2)$$

и

$$A_P^{\partial p} = \pm P^{-1}(a_{-1} + P^{-1}(a_{-2} + \dots + P^{-1}(a_{-(m-1)} + a_{-m} P^{-1}) \dots)) \quad (3)$$

Согласно этому алгоритму, перевод числа, включающего в себя целую и дробную части, состоит из вычислительных процессов двух видов:

1) последовательного деления нацело целой части и образующихся целых частных на основание новой системы счисления;

2) последовательного умножения дробной части и дробных частей получающихся произведений на то же новое основание, записанное, как и в первом случае, цифрами исходной системы счисления.

При переводе целой части числа в ходе процесса последовательного деления получают остатки, выраженные цифрами исходной системы счисления. Они представляют собой цифры a_0, a_1, \dots, a_n целой части числа, переведенного в новую систему счисления. Последний остаток является старшей цифрой переведенного числа. Процесс деления продолжается до тех пор, пока частное не станет равным нулю.

При переводе дробной части числа при каждом умножении получают целые части, которые исключаются из последующих умножений. Эти целые части, изображенные цифрами исходной системы счисления, представляют собой цифры дробной части, переведенной в новую систему счисления. Значение первой целой части является первой цифрой после запятой переведенного числа. Процесс умножения продолжаем до тех пор, пока дробная часть произведения не станет равной нулю или не выделится период.

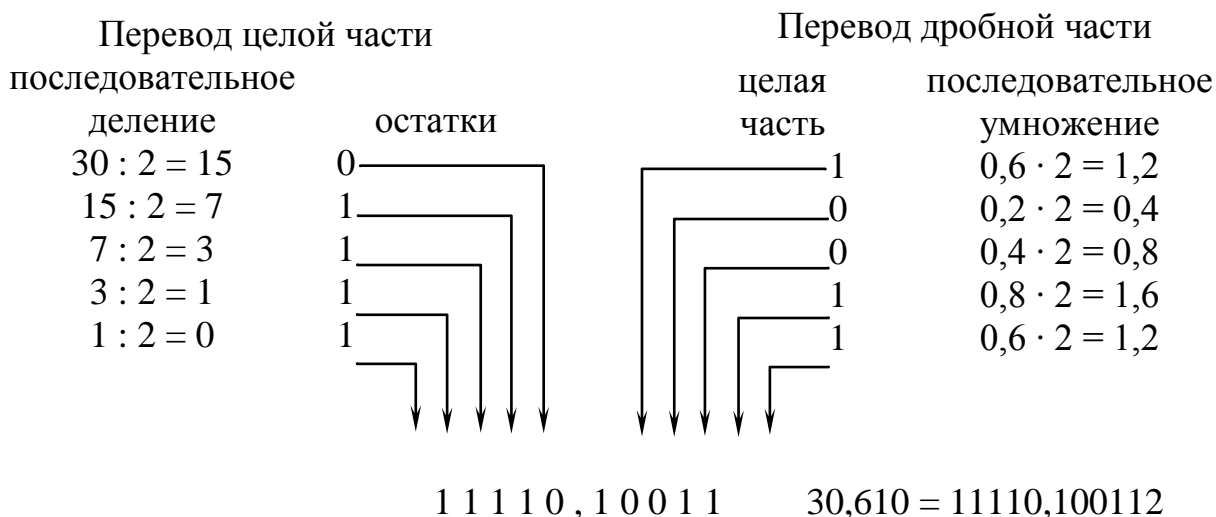
Следует отметить, что все арифметические действия осуществляются по правилам исходной системы счисления.

По универсальному алгоритму наиболее просто переводить числа из десятичной системы счисления в любую другую.

Пример 1.

Требуется перевести число 30,6 в десятичной системе счисления в двоичную систему счисления.

Согласно вышесказанному, переводим отдельно целую и дробную части



Для перевода числа из любой системы счисления в десятичную систему можно воспользоваться представлением исходного числа как суммы произведений цифр данного числа и степеней основания исходной системы счисления. Алгоритм такого перевода выглядит следующим образом (действия проводятся в десятичной системе счисления):

- 1) цифры исходного числа нумеруются справа налево начиная с нуля – для целой части числа и слева направо начиная с -1 – для дробной части;
- 2) каждая цифра числа переводится в число в десятичной системе;
- 3) десятичное число, соответствующее каждой цифре, умножается на основание исходной системы счисления в пронумерованной степени, и результаты складываются.

Пример 2.

Требуется перевести число 111101,01 в двоичной системе счисления в десятичную систему счисления:

$$1^5 1^4 1^3 1^2 0^1 1^0, 0^{-1} 1^{-2} = \\ = 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 0 \cdot 2^{-1} + 1 \cdot 2^{-2} = 61,25_{10}$$

Пример 3.

Требуется перевести число 3E5A1 в шестнадцатеричной системе счисления в десятичную систему счисления:

$$3^4 E^3 5^2 A^1 1^0_{16} = 3 \cdot 16^4 + E \cdot 16^3 + 5 \cdot 16^2 + A \cdot 16^1 + 1 \cdot 16^0 = \\ = 3 \cdot 16^4 + 14 \cdot 16^3 + 5 \cdot 16^2 + 10 \cdot 16^1 + 1 \cdot 16^0 = 255393_{10}$$

Еще более простой алгоритм применяется для работы с системами счисления, основание которых представляет собой ту или иную степень числа 2. К таким системам счисления относятся: двоичная, восьмеричная, шестнадцатеричная и т.д. Иногда подобные системы счисления называют смешанными. Для записи любой цифры числа, представленного в системе счисления, основание которой 2 в любой степени, необходимо количество двоичных цифр, равное значению данной степени. Например, для записи восьмеричного числа – три двоичные цифры, для записи шестнадцатеричного – четыре.

Для того чтобы перевести двоичное число в восьмеричную систему счисления, преобразуемое число разделяют справа налево на группы по три двоичные цифры, при этом самая левая группа может содержать меньше трех двоичных цифр. Затем каждую группу двоичных цифр выражают в виде восьмеричной цифры с учетом того, что:

$000_2 = 0_8$	$011_2 = 3_8$	$110_2 = 6_8$
$001_2 = 1_8$	$100_2 = 4_8$	$111_2 = 7_8$
$010_2 = 2_8$	$101_2 = 5_8$	

Аналогично преобразуют двоичное число в шестнадцатеричное с той лишь разницей, что преобразуемое двоичное число делят на группы по 4

двоичных цифры в каждой, поскольку для записи любой цифры шестнадцатеричного числа необходимы четыре двоичные цифры:

$0000_2 = 0_{16}$	$0100_2 = 4_{16}$	$1000_2 = 8_{16}$	$1100_2 = C_{16}$
$0001_2 = 1_{16}$	$0101_2 = 5_{16}$	$1001_2 = 9_{16}$	$1101_2 = D_{16}$
$0010_2 = 2_{16}$	$0110_2 = 6_{16}$	$1010_2 = A_{16}$	$1110_2 = E_{16}$
$0011_2 = 3_{16}$	$0111_2 = 7_{16}$	$1011_2 = B_{16}$	$1111_2 = F_{16}$

Преобразование восьмеричного или шестнадцатеричного числа в двоичное осуществляется простым переводом каждой цифры исходного числа в группу из трех (для восьмеричного числа) или четырех (для шестнадцатеричного числа) двоичных цифр.

Пример 4.

$$\begin{aligned} 1101111011_2 &= 1 \ 101 \ 111 \ 011 = 1573_8 \\ 1101111011_2 &= 11 \ 0111 \ 1011 = 37B_{16} \\ 123_8 &= 001 \ 010 \ 011 = 1010011_2 \\ A17_{16} &= 1010 \ 0001 \ 0111 = 101000010111_2 \end{aligned}$$

Арифметические операции

Арифметические действия над числами (сложение, вычитание, умножение и деление) в любой позиционной системе счисления производятся по тем же правилам, что и десятичной системе, так как все они основываются на правилах выполнения действий над соответствующими многочленами. При этом нужно только пользоваться теми таблицами сложения и умножения, которые соответствуют данному основанию b системы счисления. Операции сложения и умножения производятся по тем же правилам, что и в десятичной системе счисления.

Пример 1.

Требуется сложить два числа в двоичной системе счисления:

$$\begin{array}{r} 1 \ 0 \ 1 \ 1 \ 1 \\ + \ 1 \ 0 \ 1 \ 1 \ 0 \\ \hline 1 \ 0 \ 1 \ 1 \ 0 \ 1 \end{array}$$

Вычитание чисел в двоичной системе счисления может осуществляться двумя способами:

- 1) из большего по абсолютной величине числа вычитается меньшее, и у результата ставится соответствующий знак;
- 2) вычитаемое предварительно преобразуется в дополнительный код (перед преобразованием количество разрядов в числах выравнивается), после чего оба числа суммируются.

Для получения дополнительного кода отрицательного числа необходимо:

- 1) значения всех разрядов изменить на противоположные, т.е. все нули заменить единицами, а единицы нулями (получить обратный код исходного числа);
- 2) к полученному обратному коду прибавить единицу в младшем разряде.

Пример 2.

Требуется найти разность двух чисел в двоичной системе счисления:
 $10111_2 - 1101_2$

Первый способ:

$$\begin{array}{r} 1\ 0\ 1\ 1\ 1 \\ -\quad 1\ 1\ 0\ 1 \\ \hline 1\ 0\ 1\ 0 \end{array}$$

Второй способ:

Получим дополнительный код числа 1101_2

$1101 \Rightarrow 01101$ (добавили один пустой разряд справа)

$01101 \Rightarrow 10010$ (заменяли $0 \Leftrightarrow 1$)

$10010 \Rightarrow 10011$ (прибавили 1 в младшем разряде)

$$\begin{array}{r} 1\ 0\ 1\ 1\ 1 \\ +\ 1\ 0\ 0\ 1\ 1 \\ \hline \cancel{1}\ 0\ 1\ 0\ 1\ 0 \end{array}$$

Игнорируем 1 переноса из старшего разряда и получаем результат, равный 01010_2 .

Порядок выполнения работы

1. Ознакомиться с основными теоретическими положениями.
2. Получить у преподавателя вариант задания (см. приложение).
3. Выбрать наиболее рациональный метод перевода чисел для каждого задания.
4. Перевести числа из одной системы счисления в другую с помощью выбранного метода.

Содержание отчета

1. Цель работы.
2. Задание.
3. Перевод чисел со всеми промежуточными расчетами.
4. Выводы по работе.

Контрольные вопросы

1. Чем отличаются позиционные системы счисления от аддитивных? Приведите примеры.
2. Чем характеризуется позиционная система счисления?
3. Какие системы счисления относятся к нетрадиционным? Приведите примеры.
4. Каким образом осуществляется перевод по универсальному алгоритму?
5. Как можно перевести числа из любой системы счисления в десятичную?
6. Назовите недостатки представления чисел в двойном коде.
7. Каким образом получается дополнительный код двоичного числа?
8. Что такое экспоненциальная форма записи числа?
9. Что подразумевается под понятиями: машинное слово, полуслово и двойное слово

Варианты заданий**Вариант 1**

1. Перевести в 10-ную систему счисления:

11011001.11_2 , 5402.3_6 , 3004.6_8 , 4625_7 , $2A45.D_{16}$

2. Перевести в 2-ную систему счисления:

5617_{10} , 20341.25_8 , $A1DF4.42_{16}$

3. Перевести в 8-ную систему счисления:

354.6_{10} , 101000111.01101_2 , $E246.2CC_{16}$

4. Перевести в 16-ную систему счисления:

768.25_{10} , 1357.653_8 , 1010011110.1010_2

5. Перевести в 3-ную систему счисления:

45.24_{10} , 224.34_5

6. Перевести в 7-ную систему счисления:

1302.23_4

7. Выполнить следующие действия:

$11010101_2 + 1110_2$ $1001110101_2 + 10111101_2$

$11011011_2 - 110101110_2$ $110001010_2 - 101101001_2$

Вариант 2

1. Перевести в 10-ную систему счисления:

10101101.11_2 , 35602.3_7 , 32301.1_4 , 5043_8 , $1C67.EF_{16}$

2. Перевести в 2-ную систему счисления:

74210_{10} , 5301.72_8 , $6A1DE9.6_{16}$

3. Перевести в 8-ную систему счисления:

781.24_{10} , 1110010.1101_2 , $AC9F.4CF_{16}$

4. Перевести в 16-ную систему счисления:

1825.26_{10} , 3401.541_8 , 100100111.1110_2

5. Перевести в 4-ную систему счисления:

606.53_{10} , 36042.42_7

6. Перевести в 9-ную систему счисления:

$A54C.6F_{16}$

7. Выполнить следующие действия:

$1011101_2 + 101101_2$ $1011010101_2 + 1010101_2$

$111011011_2 - 110101110_2$ $110001010_2 - 10101001_2$

Вариант 3

1. Перевести в 10-ную систему счисления:

10110001.101_2 , 35404.04_6 , 3202.201_5 , 43761_8 , 1507.32_8

2. Перевести в 2-ную систему счисления:

65078_{10} , 20134.24_8 , $1AD9E.5_{16}$

3. Перевести в 8-ную систему счисления:

2646.52_{10} , 10100110.0111_2 , $90135.C45_{16}$

4. Перевести в 16-ную систему счисления:

4032.3_{10} , 35471.3621_8 , 1010001101.10110_2

5. Перевести в 6-ную систему счисления:

4071.73_{10} , 2013.14_5

6. Перевести в 5-ную систему счисления:

504107.54_8

7. Выполнить следующие действия:

$10011101_2 + 1011001_2$ $1010101011_2 + 11010101_2$

$11011011_2 - 10101110_2$ $11001010_2 - 1011001_2$

Вариант 4

1. Перевести в 10-ную систему счисления:

10101011.1_2 , 3024.3_7 , 3442.04_5 , 204.31_6 , 472.2_8

2. Перевести в 2-ную систему счисления:

23103_{10} , 1305.41_8 , $FC2CD.E_{16}$

3. Перевести в 8-ную систему счисления:

4317.4_{10} , 10110010.01101_2 , $67A96C.571F_{16}$

4. Перевести в 16-ную систему счисления:

5413.45_{10} , 13461.552_8 , 1111001101.101_2

5. Перевести в 7-ную систему счисления:

5162.85_{10} , 2023.02_4

6. Перевести в 9-ную систему счисления:

42021.23_8

7. Выполнить следующие действия:

$110101_2 + 1011001_2$ $10101011_2 + 10110101_2$

$101011011_2 - 100111101_2$ $110100010_2 - 1011001_2$

Вариант 5

1. Перевести в 10-ную систему счисления:
 1011101.1001_2 , 53024.1_9 , 1243.5_8 , 1062.42_{16} , 3034.22_5
2. Перевести в 2-ную систему счисления:
 31202_{10} , 5601.46_8 , $23AB.7_{16}$
3. Перевести в 8-ную систему счисления:
 2005.44_{10} , 10111110.00101_2 , $625BC.F_{16}$
4. Перевести в 16-ную систему счисления:
 876.92_{10} , 1652.312_8 , 11110101101.101_2
5. Перевести в 5-ную систему счисления:
 16402.85_{10} , 4423.02_7
6. Перевести в 7-ную систему счисления:
 42021.23_5
7. Выполнить следующие действия:
 $1101001_2 + 1011001_2$ $101011011_2 + 10110101_2$
 $10011011_2 - 100111101_2$ $110001010_2 - 1011101_2$

Вариант 6

1. Перевести в 10-ную систему счисления:
 1010001.001_2 , 8024.1_9 , 1253.5_6 , 1092.42_{16} , 3034.22_7
2. Перевести в 2-ную систему счисления:
 7203_{10} , 7012.62_8 , $D245.7E_{16}$
3. Перевести в 8-ную систему счисления:
 6402.45_{10} , 100101110.00101_2 , $9B14C.F5_{16}$
4. Перевести в 16-ную систему счисления:
 934.62_{10} , 6543.15_8 , 1101011101.10101_2
5. Перевести в 9-ную систему счисления:
 30412.44_{10} , 4021.32_5
6. Перевести в 7-ную систему счисления:
 67321.73_8
7. Выполнить следующие действия:
 $11001101_2 + 1001001_2$ $111011011_2 + 10110111_2$
 $100101111_2 - 110111101_2$ $1100010101_2 - 101110101_2$

Вариант 7

1. Перевести в 10-ную систему счисления:

1101101.0101_2 , 6612.4_8 , 10403.2_5 , $10B44.26_{16}$, 4421.56_9

2. Перевести в 2-ную систему счисления:

5725_{10} , 1342.52_8 , $5FF.2D4_{16}$

3. Перевести в 8-ную систему счисления:

3418.42_{10} , 10011101110.10101_2 , $C458.45A_{16}$

4. Перевести в 16-ную систему счисления:

934.74_{10} , 1223.165_8 , 10111011101.101001_2

5. Перевести в 3-ную систему счисления:

245.42_{10} , 3021.32_4

6. Перевести в 5-ную систему счисления:

$44A.13_{16}$

7. Выполнить следующие действия:

$10101101_2 + 10011001_2$ $1011011011_2 + 101110111_2$

$1001010111_2 - 1101111101_2$ $110001011_2 - 10110101_2$

Вариант 8

1. Перевести в 10-ную систему счисления:

110110011.11001_2 , 1202.1_3 , 3212.2_4 , $2FA3.F4_{16}$, 3541.302_6

2. Перевести в 2-ную систему счисления:

32074_{10} , 123452.502_8 , $5A67.274_{16}$

3. Перевести в 8-ную систему счисления:

5610.63_{10} , 100101110.101101_2 , 8651.4053_{16}

4. Перевести в 16-ную систему счисления:

43607.25_{10} , 120253.13_8 , 10110110101.101011_2

5. Перевести в 4-ную систему счисления:

233.42_{10} , 3031.22_6

6. Перевести в 5-ную систему счисления:

3210.12_4

7. Выполнить следующие действия:

$1010101_2 + 10011001_2$ $1011011011_2 + 10110111_2$

$100101011_2 - 1101111101_2$ $1100011011_2 - 101010101_2$

Вариант 9

1. Перевести в 10-ную систему счисления:
 111001010.1101_2 , 4302.4_5 , 65203.4_7 , $5BB.D2_{16}$, 33210.112_4
2. Перевести в 2-ную систему счисления:
 5406_{10} , 54752.032_8 , 589003.9402_{16}
3. Перевести в 8-ную систему счисления:
 4328.78_{10} , 1001000110.101001_2 , $A341.C34_{16}$
4. Перевести в 16-ную систему счисления:
 66201.45_{10} , 700120.103_8 , 1011011110101.1000101_2
5. Перевести в 3-ную систему счисления:
 253.42_{10} , 3204.4_5
6. Перевести в 4-ную систему счисления:
 20211.2_4
7. Выполнить следующие действия:
 $10101001_2 + 10011001_2$
 $1001011011_2 + 110110111_2$
 $100101011_2 - 110111101_2$
 $1100001011_2 - 1010101011_2$

Вариант 10

1. Перевести в 10-ную систему счисления:
 11101010.01_2 , 553.4_6 , 20301.4_5 , 54402.23_8 , 4332410.14_5
2. Перевести в 2-ную систему счисления:
 6700_{10} , 34002.312_8 , $80024003.F402_{16}$
3. Перевести в 8-ную систему счисления:
 3508.65_{10} , 1001011001010.1011001_2 , $B3A4.D_{16}$
4. Перевести в 16-ную систему счисления:
 71201.36_{10} , 3062320.5036_8 , 101100010101.01101_2
5. Перевести в 9-ную систему счисления:
 25320.32_{10} , 6234.36_7
6. Перевести в 6-ную систему счисления:
 442201.2_5
7. Выполнить следующие действия:
 $1001111_2 + 1000101_2$
 $100101011_2 + 110110111_2$
 $1001011011_2 - 1101110101_2$
 $11000011011_2 - 10110101011_2$

Вариант 11

1. Перевести в 10-ную систему счисления:

1101010.011_2 , 33102.2_4 , 20101.1_3 , 354502.21_6 , 4332410.14_5

2. Перевести в 2-ную систему счисления:

6061_{10} , 346102.5152_8 , $A4CC24703.F02B_{16}$

3. Перевести в 8-ную систему счисления:

3508.52_{10} , 111001000101.10111001_2 , 22084.89_{16}

4. Перевести в 16-ную систему счисления:

2305.64_{10} , 62237.5076_8 , 1010010101.011101_2

5. Перевести в 3-ную систему счисления:

26360.42_{10} , 4620.46_7

6. Перевести в 7-ную систему счисления:

1011001.2_3

7. Выполнить следующие действия:

$10011011_2 + 10010101_2$

$1011101011_2 + 110110111_2$

$10011011011_2 - 11011110101_2$

$1001000011011_2 - 10110101011_2$

Вариант 12

1. Перевести в 10-ную систему счисления:

11011010.1011_2 , 34001.2_6 , 22301.1_4 , 56236.24_7 , 34304.24_5

2. Перевести в 2-ную систему счисления:

7382_{10} , 54703.465_8 , $B32CD.5634C_{16}$

3. Перевести в 8-ную систему счисления:

925.56_{10} , 110010010101.10111101_2 , $2B08E.8A_{16}$

4. Перевести в 16-ную систему счисления:

7538.62_{10} , 86227.50376_8 , 101000101.010101_2

5. Перевести в 5-ную систему счисления:

46560.42_{10} , 53601.36_7

6. Перевести в 9-ную систему счисления:

45222.4_6

7. Выполнить следующие действия:

$100110011_2 + 101010101_2$

$10110101011_2 + 1101010111_2$

$101011011011_2 - 110111101101_2$

$10010100011011_2 - 10110101011_2$

Вариант 13

1. Перевести в 10-ную систему счисления:
 11010011.1101_2 , 30501.4_6 , 23303.1_5 , 8341.54_9 , 1301.22_4
2. Перевести в 2-ную систему счисления:
 6252_{10} , 34603.405_8 , $3C5D.5344C_{16}$
3. Перевести в 8-ную систему счисления:
 5972.76_{10} , 111010010101.10101101_2 , $2AA.44F_{16}$
4. Перевести в 16-ную систему счисления:
 53809.45_{10} , 6723.53376_8 , 1010100101.0101_2
5. Перевести в 6-ную систему счисления:
 45672.21_{10} , 23201.3_4
6. Перевести в 8-ную систему счисления:
 23562.4_7
7. Выполнить следующие действия:
 $10010011_2 + 1010101_2$
 $10111101011_2 + 1101010111_2$
 $101011011011_2 - 110111101001_2$
 $100101000011011_2 - 10110101011_2$

Вариант 14

1. Перевести в 10-ную систему счисления:
 110100101.1011_2 , 2021.2_3 , 76602.34_8 , 56441.32_7 , 1442.23_5
2. Перевести в 2-ную систему счисления:
 53311_{10} , 340610.4022_8 , $A3352.5444_{16}$
3. Перевести в 8-ную систему счисления:
 6532.65_{10} , 10010011101.101101_2 , $43CA.567_{16}$
4. Перевести в 16-ную систему счисления:
 54377.25_{10} , 267023.54376_8 , 10101010101.10101_2
5. Перевести в 7-ную систему счисления:
 474232.23_{10} , 11231.5_6
6. Перевести в 3-ную систему счисления:
 255752.2_8
7. Выполнить следующие действия:
 $100101011_2 + 1010101_2$
 $10111101011_2 + 11010111_2$
 $10101011011_2 - 11011101001_2$
 $10001000011011_2 - 1010101011_2$

Вариант 15

1. Перевести в 10-ную систему счисления:
 110110101.11_2 , 121.32_4 , 5730.45_8 , 4342.03_5 , 65402.23_7
2. Перевести в 2-ную систему счисления:
 530204_{10} , 10457.2632_8 , $D452.5C_{16}$
3. Перевести в 8-ную систему счисления:
 6045.53_{10} , 1001101101.11101_2 , $3AF.623_{16}$

4. Перевести в 16-ную систему счисления:
 48307.205_{10} , 5004523.576_8 , 101101010101.101101_2
5. Перевести в 9-ную систему счисления:
 55302.23_{10} , 102301.5_6
6. Перевести в 5-ную систему счисления:
 $2A51.3_{16}$
7. Выполнить следующие действия:
 $1001011011_2 + 1010101_2$
 $10111101011_2 + 110110111_2$
 $101010111011_2 - 110111101001_2$
 $100010001011_2 - 101010111_2$

Вариант 16

1. Перевести в 10-ную систему счисления:
 1100101.101_2 , 1231.02_4 , 5440.21_6 , 1002.21_3 , 430212.43_5
2. Перевести в 2-ную систему счисления:
 2054_{10} , 533101.23_8 , $2DD.56_{16}$
3. Перевести в 8-ную систему счисления:
 493.21_{10} , 1001101.1101_2 , $106A.403_{16}$
4. Перевести в 16-ную систему счисления:
 48301.223_{10} , 5002372.502_8 , 10110100101.10111_2
5. Перевести в 6-ную систему счисления:
 32201.03_{10} , 3014.05_6
6. Перевести в 16-ную систему счисления:
 6251.3_7
7. Выполнить следующие действия:
 $10010111_2 + 1010101_2$
 $1011110011_2 + 11011011_2$
 $10101011011_2 - 11011101001_2$
 $10001001011_2 - 101010111_2$

Вариант 17

1. Перевести в 10-ную систему счисления:

1101101.01_2 , 5231.25_6 , 20041.41_7 , 4024.11_5 , $F2C12.5_{16}$

2. Перевести в 2-ную систему счисления:

6204_{10} , 31201.603_8 , $2C708.A6_{16}$

3. Перевести в 8-ную систему счисления:

4903.52_{10} , 10101101.01101_2 , $106F.5281_{16}$

4. Перевести в 16-ную систему счисления:

4091.52_{10} , 230472.202_8 , 10110110101.101011_2

5. Перевести в 5-ную систему счисления:

7201.43_{10} , 1024.13_7

6. Перевести в 6-ную систему счисления:

103201.3_4

7. Выполнить следующие действия:

$11010111_2 + 1010101_2$

$1011001011_2 + 11011_2$

$1010101111_2 - 1101101001_2$

$100010010101_2 - 1010101011_2$

Вариант 18

1. Перевести в 10-ную систему счисления:

1100101.01_2 , 1031.23_4 , 20501.421_6 , 3024.11_8 , $1A2.51_{16}$

2. Перевести в 2-ную систему счисления:

7041_{10} , 26601.401_8 , $21A07.C6_{16}$

3. Перевести в 8-ную систему счисления:

6503.32_{10} , 110001101.1001101_2 , $106B.891_{16}$

4. Перевести в 16-ную систему счисления:

1760.22_{10} , 51202.202_8 , 10110101.10101_2

5. Перевести в 5-ную систему счисления:

2061.45_{10} , 1452.53_7

6. Перевести в 7-ную систему счисления:

12401.2_6

7. Выполнить следующие действия:

$110101101_2 + 10100101_2$

$10110011011_2 + 1101001_2$

$10101010111_2 - 11011011001_2$

$100010010101_2 - 101011011_2$

Вариант 19

1. Перевести в 10-ную систему счисления:

11100101.01_2 , 30341.23_5 , 2201.21_6 , 6024.14_8 , $C24.74_{16}$

2. Перевести в 2-ную систему счисления:

6654_{10} , 54621.434_8 , $AAA07.86_{16}$

3. Перевести в 8-ную систему счисления:

7503.36_{10} , 1011101101.100101_2 , $30FB.8C_{16}$

4. Перевести в 16-ную систему счисления:

1240.52_{10} , 531602.702_8 , 101110111.101011_2

5. Перевести в 5-ную систему счисления:

20431.34_{10} , 6322.13_7

6. Перевести в 7-ную систему счисления:

42603.2_6

7. Выполнить следующие действия:

$1101011101_2 + 101100101_2$

$101100101011_2 + 1101001_2$

$10101010111_2 - 110110111001_2$

$1000101010101_2 - 1010111011_2$

Вариант 20

1. Перевести в 10-ную систему счисления:

10110011.01_2 , 40221.31_6 , 2301.31_4 , 7204.14_8 , $204.F4_{16}$

2. Перевести в 2-ную систему счисления:

5034_{10} , 22001.214_8 , $F07.86_{16}$

3. Перевести в 8-ную систему счисления:

6073.36_{10} , 101101.0101_2 , $B1B.AA_{16}$

4. Перевести в 16-ную систему счисления:

4220.52_{10} , 10602.02_8 , 10100111.10011_2

5. Перевести в 6-ную систему счисления:

2314.41_{10} , 222.17_8

6. Перевести в 5-ную систему счисления:

42003.02_{16}

7. Выполнить следующие действия:

$1101011101_2 + 10100101_2$

$10110101011_2 + 1101001_2$

$101001010111_2 - 110110111001_2$

$10001010110101_2 - 10110111011_2$

Вариант 21

1. Перевести в 10-ную систему счисления:
 101101.01_2 , 40231.31_7 , 230.31_6 , 1204.14_9 , $20C4.D_{16}$
2. Перевести в 2-ную систему счисления:
 40014_{10} , 2430.04_8 , $CCF.26_{16}$
3. Перевести в 8-ную систему счисления:
 6102.3_{10} , 1000101.0011_2 , $34B.41_{16}$
4. Перевести в 16-ную систему счисления:
 4245.54_{10} , 12773.32_8 , 110100111.101011_2
5. Перевести в 4-ную систему счисления:
 7724.61_{10} , 11012.12_3
6. Перевести в 3-ную систему счисления:
 4503.52_6
7. Выполнить следующие действия:
 $110011101_2 + 10100101_2$
 $1011001011_2 + 1101001_2$
 $10100010111_2 - 110110111001_2$
 $1000101010101_2 - 10110111011_2$

Вариант 22

1. Перевести в 10-ную систему счисления:
 1001100.11_2 , 2314.34_5 , 2430.31_8 , 6604.64_7 , $F0C4.C_{16}$
2. Перевести в 2-ную систему счисления:
 4452_{10} , 3200.21_8 , $325CF.25_{16}$
3. Перевести в 8-ную систему счисления:
 6305.33_{10} , 10100101.0011_2 , $3DD.451_{16}$
4. Перевести в 16-ную систему счисления:
 6435.24_{10} , 7573.12_8 , 11010011.10101_2
5. Перевести в 4-ную систему счисления:
 8724.61_{10} , 10011.1_2
6. Перевести в 3-ную систему счисления:
 484.62_9
7. Выполнить следующие действия:
 $11001101_2 + 10100101_2$
 $101100101_2 + 1101001_2$
 $101000100111_2 - 1101010111001_2$
 $1000101011101_2 - 10110111011_2$

Вариант 23

1. Перевести в 10-ную систему счисления:
 101101.01_2 , 2314.25_6 , 20451.45_7 , 4011.41_8 , $C121.6_{16}$
2. Перевести в 2-ную систему счисления:
 3304_{10} , 34401.604_8 , $2C881.666_{16}$
3. Перевести в 8-ную систему счисления:
 9803.42_{10} , 10100110101.010101_2 , $106F.5281_{16}$
4. Перевести в 16-ную систему счисления:
 4091.52_{10} , 230472.202_8 , 10110110101.111011_2
5. Перевести в 16-ную систему счисления:
 73101.4_{10} , 3444.14_5
6. Перевести в 6-ную систему счисления:
 10011101.1001_3
7. Выполнить следующие действия:
 $110111_2 + 1010101_2$
 $10101001011_2 + 110101_2$
 $10101011110_2 - 11011010011_2$
 $1000100101010_2 - 10101010101_2$

Вариант 24

1. Перевести в 10-ную систему счисления:
 11001.01_2 , 3451.533_6 , 5521.3_6 , 1324.13_8 , $C82.9_{16}$
2. Перевести в 2-ную систему счисления:
 70221_{10} , 74601.341_8 , $B07D.68_{16}$
3. Перевести в 8-ную систему счисления:
 6603.32_{10} , 1101001101.101101_2 , $1A06.89D_{16}$
4. Перевести в 16-ную систему счисления:
 1330.25_{10} , 33202.202_8 , 101000101.101101_2
5. Перевести в 16-ную систему счисления:
 8861.41_{10} , 33142.53_7
6. Перевести в 3-ную систему счисления:
 1401.1_5
7. Выполнить следующие действия:
 $1101011101_2 + 110100101_2$
 $101100111011_2 + 11101001_2$
 $101011010111_2 - 110111011001_2$
 $1000100110101_2 - 1010111011_2$

Вариант 25

1. Перевести в 10-ную систему счисления:

110101.001_2 , 34401.4_5 , 20323.1_4 , 8241.5_9 , 3201.2_7

2. Перевести в 2-ную систему счисления:

1152_{10} , 3433603.105_8 , $885D.24C2_{16}$

3. Перевести в 8-ную систему счисления:

8912.7_{10} , 1110010101.1001101_2 , $23A.14_{16}$

4. Перевести в 16-ную систему счисления:

5109.41_{10} , 3423.566_8 , 101110100101.01101_2

5. Перевести в 6-ную систему счисления:

4522.24_{10} , 1201.2_4

6. Перевести в 8-ную систему счисления:

2620.6_7

7. Выполнить следующие действия:

$100101011_2 + 10101101_2$

$1011111011_2 + 110100111_2$

$10111011011_2 - 11011101001_2$

$10010100011011_2 - 1011011011_2$

Практическое занятие №2

Работа с файлами и директориями в командной строке

Цель работы: изучить функционирование операционной системы MS DOS и приобретение практических навыков работы в ней с файлами и директориями при помощи основных команд.

Теоретическая часть

Основные понятия. Назначение операционной системы

При работе пользователя на компьютере часто возникает необходимость выполнить операции с прикладной программой в целом, организовать работу внешних устройств, проверить работу различных блоков, скопировать информацию и т.п. Из всего многообразия подобных операций выделяют типовые и реализуют их с помощью специализированных программ, принимаемых в качестве стандартных средств и поставляемых вместе с аппаратной частью.

Программы, организующие работу устройств и не связанные со спецификой решаемой задачи, входят в состав комплекса программ, называемого операционной системой (ОС).

ОС – совокупность программных средств, обеспечивающая управление аппаратной частью компьютера и прикладными программами, а также их взаимодействие между собой и пользователем.

ОС образует автономную среду, не связанную ни с одним из языков программирования. Любая же прикладная программа связана с ОС и может эксплуатироваться только на тех компьютерах, где имеется аналогичная системная среда. Для работы с ОС необходимо овладеть языком этой среды – совокупностью команд, структура которых определяется синтаксисом этого языка.

ОС выполняет следующие функции:

- управление работой каждого блока компьютера и их взаимодействием;
- управление выполнением программ;
- организацию хранения информации во внешней памяти;
- взаимодействие пользователя с компьютером, т.е. поддержка интерфейса пользователя.

Обычно ОС хранится на жестком диске или на специальном гибком диске, называемом системным. При включении компьютера ОС автоматически загружается с диска в оперативную память и занимает в ней определенное место. ОС создается не для отдельной модели компьютера, а для серии компьютеров.

DOS (ДОС – дисковая ОС) – это имя, используемое фирмой IBM для обозначения основной ОС, которая работает на персональных компьютерах (ПК) семейства IBM PC. DOS была создана для IBM фирмой Microsoft, одной

из ведущих фирм по производству программного обеспечения для ПК. Фирма Microsoft также создала версии DOS для компьютеров других семейств. Эти другие версии обычно называют MS DOS (сокращение от Microsoft DOS).

Понятие файла

В основе любой ОС лежит принцип организации работы внешнего устройства для хранения информации. Несмотря на то, что внешняя память может быть реализована на разных материальных носителях, их объединяет принятый в ОС принцип организации хранения логически связанных наборов информации в виде так называемых файлов.

Файл – логически связанная совокупность данных, для размещения которой во внешней памяти выделяется именованная область.

Файл служит учетной единицей информации в ОС. Любые действия с информацией в MS DOS осуществляются над файлами: запись на диск, вывод на экран, ввод с клавиатуры, печать и прочее.

В файлах могут храниться разнообразные виды и формы представления информации: тексты, рисунки, чертежи, числа, программы, таблицы и т.п. Особенности конкретных файлов определяются форматом, под которым понимается элемент языка, в символическом виде описывающий представление информации в файле.

Для характеристики файла используются следующие параметры:

- полное имя файла;
- объем файла в байтах;
- дата создания файла;
- время создания файла;
- специальные атрибуты файла: R – только для чтения, H – скрытый файл, S – системный файл, A – архивированный файл.

С понятием файла в MS DOS тесно связано понятие логического диска. Логический диск создается и управляется специальной программой (драйвером). Он имеет уникальное имя, например, C, D, E, F. Логический диск может быть организован на жестком, гибком, лазерном дисках, в оперативной памяти (электронный диск) и т.п. На одном физическом диске может быть создано несколько логических дисков. В дальнейшем изложении под диском будем понимать логический диск.

Способы обращения к файлу

К файлу можно обращаться с помощью имени или с помощью полного имени. Рассмотрим эти варианты.

Правило образования имени. Имя файла уникально и служит для отличия одного файла от другого. Имя файла в MS DOS образуется не более чем из восьми символов, причем используются только буквы латинского алфавита и цифры, а первым символом обязательно должна быть буква.

В качестве имени файла можно использовать символьное имя устройства:

- PRN или LPT1 (2, 3) – принтер или любое устройство, подключенное к параллельному порту;
- CON – консоль (клавиатура при вводе и дисплей при выводе);
- COM1 (2, 3, 4) – какое-либо внешнее устройство, подключенное к последовательному порту;
- NUL – фиктивное устройство; вывод в файл NUL никуда не направляется, а просто уничтожается.

Правило образования полного имени.

Полное имя более подробно характеризует файл и образуется из имени файла и типа (расширения), разделенных точкой.

Тип файла служит для характеристики хранящейся в файле информации и образуется не более чем из трех символов, причем используются (как и при образовании имени) только буквы латинского алфавита.

Пример. PRIMER.PAS – файл PRIMER для хранения про-грамм на Паскале; STRAN.TXT – файл STRAN для хранения текста; COPY.COM – файл COPY, содержащий программу ОС по копированию файлов.

При работе на ПК установлен ряд соглашений по заданию типа файла. Часть таких соглашений приведена в таблице 1.

Таблица 1

Соглашения по типу файлов

Тип	Назначение
.ARJ	Архивный файл
.BAK	Копия файла, создаваемая при перезаписи файла
.BAT	Командный файл
.COM	Командный системный файл, исполняемый файл
.DAT	Файл данных
.EXE	Исполняемый файл
.HLP	Файл для справочной информации
.LIB	Библиотека программ
.PIC	Данные выводимого на экран изображения
.PRN	Листинг (распечатка программы)
.SYS	Файлы, расширяющие возможности ОС (драйверы)
.TXT	Текстовый файл

Часто возникает ситуация, когда надо работать не с одним файлом, а с группой файлов. Например, необходимо выполнить операции: копирование группы файлов с одного диска на другой; удаление группы файлов; перемещение группы файлов на другой диск; поиск группы файлов заданного типа и т.п.

Такие операции легко выполнять, пользуясь при формировании имен и типов файлов шаблоном.

Шаблон имени файла – специальная форма, в которой в полях имени и типа файла используются символы '*' и '?'.

Символ '*' служит для замены любой последовательности символов. В шаблоне может быть использовано в поле имени и типа по одному символу '*'.

Пример. Задав имя '*.TXT', вы обратитесь ко всем текстовым файлам. Задав имя 'SD*.*', вы обратитесь ко всем файлам, имя которых начинается с букв SD.

Символ '?' служит для замены одного символа на месте, где стоит вопросительный знак. В шаблоне может быть использовано несколько таких символов.

Пример. Имя 'RT??.PAS' позволит обратиться ко всем файлам типа PAS, имя которых состоит из четырех символов, причем первые два символа обязательно RT, а третий и четвертый – любые.

Характеристика MS DOS. Организация доступа к файлу

Способ хранения файлов на диске и организацию доступа к ним можно сравнить соответственно с организацией хранения книг в библиотеке и процедурой поиска нужной книги по ее шифру из каталога.

Доступ – процедура установления связи с памятью и размещенным в ней файлом для записи и чтения данных.

Директория (каталог) – справочник (список) файлов с указанием месторасположения на диске.

Различают два состояния директории – активное (текущее) и пассивное. MS DOS помнит текущую директорию на каждом логическом диске.

Текущая директория – это директория, в которой работа пользователя производится в текущее машинное время.

Пассивная директория – это директория, с которой в данный момент времени не имеется связи.

В ОС MS DOS принята иерархическая структура организации директорий (каталогов). На каждом диске всегда имеется главная (корневая) директория. Она находится на нулевом (высшем) уровне иерархии и обозначается символом '\'. Корневая директория создается при форматировании (инициализации, разметке) диска и не может быть удалена средствами MS DOS. В корневую директорию могут входить другие директории (директории первого уровня) и файлы, которые создаются и удаляются командами ОС. В свою очередь, в директории первого уровня могут входить поддиректории (директории второго уровня) и т.д.

Родительская директория – это директория, имеющая поддиректории.

Поддиректория – это директория, которая входит в другую директорию.

Как правило, употребляют термин "директория" ("каталог"), подразумевая поддиректорию (подкаталог) или родительскую директорию (родительский каталог) в зависимости от контекста.

Правила наименования директорий такие же, как и правила наименования файлов. Для формального отличия от файлов обычно директориям присваивают только имена.

Доступ к содержимому файла организован из главной директории через цепочку соподчиненных директорий n-го уровня. В директориях любого уровня могут храниться записи как о файлах, так и о директориях нижнего уровня.

Описанный принцип организации доступа к файлу через директорию является основой файловой системы.

Файловая система – часть ОС, управляющая размещением и доступом к файлам и директориям на диске.

С понятием файловой системы связано понятие файловой структуры диска, под которой понимают, как размещаются на диске директории, файлы, ОС, а также какие для них выделены объемы памяти.

Доступ к файлу можно организовать следующим образом:

- если имя файла зарегистрировано в текущей директории, то достаточно указать только его имя (полное имя);
- если имя файла зарегистрировано в пассивной директории, то, находясь в текущей директории, нужно указать еще и путь.

Путь – цепочка соподчиненных директорий, которую необходимо пройти по иерархической структуре к директории, где зарегистрирован искомый файл.

При задании пути имена директорий записываются в порядке следования и отделяются друг от друга символом '\'.

Взаимодействие пользователя с ОС осуществляется с помощью командной строки, индицируемой на экране дисплея. В начале командной строки всегда имеется приглашение, которое заканчивается символом '>'. В приглашении может быть отражено: имя текущего диска, имя текущей директории, символы-разделители, текущее время и дата, путь.

Приглашение ОС – индикация на экране дисплея информации, означающей готовность ОС к вводу команд пользователя.

Возможны три варианта организации пути доступа к файлу в зависимости от места его расположения. Файл находится в текущей директории (путь отсутствует). При организации доступа к файлу достаточно указать его полное имя.

1. Файл находится в пассивной директории одного из нижних уровней, подчиненного текущей директории. При организации доступа к файлу необходимо указать путь, в котором перечислены имена всех директорий нижнего уровня, лежащих на этом пути (включая директорию, в которой находится данный файл).

2. Файл находится в пассивной директории на другой ветке по отношению к местонахождению текущей директории. Здесь необходимо указать путь, начиная с корневой директории, то есть с символа '\'. Горизонтальные переходы из директории в директорию недопустимы.

Модульная структура MS DOS. Модули ОС MS DOS

Понятие модуля широко используется применительно как к аппаратной, так и к программной части компьютера.

Модуль – унифицированная самостоятельная функциональная часть системы, имеющая законченное оформление и средства сопряжения с другими функциональными узлами и модулями.

Структуру ОС MS DOS образуют следующие модули:

- 1) BIOS (Basic Input/Output System) – базовая система ввода-вывода;
- 2) модуль расширения – EM BIOS (Extension Module BIOS) в виде файла с именем IO.SYS;
- 3) системный загрузчик (SB – System Bootstrap);
- 4) внешние драйверы – файлы с расширением .COM, .EXE, .SYS;
- 5) базовый модуль (BM – Basic Module) в виде файла с именем MSDOS.SYS;
- 6) командный процессор или интерпретатор команд (CI – Command Interpreter) в виде файла с именем COMMAND.COM;
- 7) внешние команды, утилиты – файлы с расширением .COM, .EXE, .SYS;
- 8) инструментальные средства DOS: система программирования MS DOS QBASIC; текстовый редактор MS DOS EDITOR; отладчик DEBUG для тестирования и отладки исполняемых файлов.

Первые четыре модуля составляют машинозависимую часть ОС, а последние четыре модуля – машинонезависимую часть ОС.

Система прерываний

Основным механизмом функционирования MS DOS является система прерываний.

Прерывания – это процедуры, которые компьютер вызывает для выполнения определенной задачи.

Различают аппаратные, логические и программные прерывания.

Аппаратные прерывания инициируются аппаратурой, например, сигналом от принтера, нажатием клавиши на клавиатуре, сигналом от таймера и т.д.

Логические прерывания возникают при нестандартных ситуациях в работе микропроцессора, например, деление на ноль, переполнение регистров и т.д.

Программные прерывания инициируются программами, т.е. возникают, когда программа ждет получения сервиса со стороны другой программы, например, доступ к определенным аппаратным средствам.

Функции и назначение базовой системы ввода-вывода

BIOS находится в постоянной памяти, которая входит в комплект поставки ПК. Тип ОС может изменяться, а BIOS остается постоянным.

BIOS устанавливает связь между техническими средствами и стандартизированным программным обеспечением (ПО), а именно ОС. BIOS

содержит специальные программы (драйверы) по управлению работой стандартными внешними устройствами. Назначение BIOS состоит в выполнении наиболее простых и универсальных функций ОС, связанных с вводом-выводом. BIOS содержит также: тест функционирования ПК, проверяющий работу памяти и устройств после включения питания, программу загрузки ОС. BIOS – общая (неизменяемая) часть всех ОС для данной модели ПК. Системный загрузчик считывает в оперативную память модуль расширения BIOS и модуль обработки прерываний.

Функции и назначение модуля расширения BIOS.

Модуль расширения BIOS придает гибкость ОС, позволяет управлять с ее помощью набором аппаратных средств ПК. Этот модуль можно модифицировать с учетом необходимых нужд конкретной версии MS DOS.

Модуль позволяет перекрыть функции BIOS в постоянном запоминающем устройстве и обеспечивает возможность подключения дополнительных драйверов (программ обслуживания внешних устройств). Основная функция этого модуля – это увеличение возможностей BIOS.

Функции и назначение базового модуля

Основная функция базового модуля – управление ресурсами ПК, файловой системой, работой программ при помощи системы прерываний. Функциями базового модуля на этапе загрузки являются: считывание в память и запуск командного процессора, инициализация векторов прерываний верхнего уровня.

Функции и назначение командного процессора

Командный процессор на диске может занимать любое место и, по сути, представляет собой выполняемую программу. Командный процессор выполняет в ПК следующие функции:

- прием и разбор команд с клавиатуры или из командного файла;
- выполнение команд MS DOS, находящихся внутри файла COMMAND.COM;
- загрузка и выполнение внешних команд MS DOS (утилит) и прикладных программ, хранящихся в виде файлов с расширением COM и EXE.

Программы с расширением COM не требуют настройки адресов после их загрузки в оперативную память, а с расширением EXE – настраиваются по месту размещения (для них задаются соответствующие адреса сегментов).

При загрузке в оперативную память командный процессор распадается на две части:

- резидентную, постоянно размещаемую в оперативной памяти;
- нерезидентную (транзитную), периодически изменяемую путем передачи данных между оперативной памятью и диском.

Резидентная часть содержит подпрограммы стандартной обработки прерываний. Здесь же находятся: программа подзагрузки нерезидентной

части в оперативную память и подпрограмма, обрабатывающая файл AUTOEXEC.BAT при запуске ПК.

Назначение загрузчика

Загрузчик BOOT RECORD (модуль начальной загрузки) всегда размещается на диске в нулевом секторе. Основное назначение загрузчика – поиск и перезапись (загрузка) с диска в оперативную память двух файлов IO.SYS и MSDOS.SYS, а также запуск модуля расширения базовой системы ввода-вывода.

Утилиты, внешние команды и драйверы

Утилиты – обслуживающие программы, поставляемые вместе с ОС в виде файлов и предоставляющие пользователю сервисные услуги (форматирование дисков, проверку дисков и т.д.).

Внешней командой принято считать программу, выдающую пользователю ряд простых запросов или выполняющуюся автоматически без специально организованного интерфейса пользователя. MS DOS имеет определенный перечень внешних команд.

Внешние драйверы – программы, дополняющие систему ввода-вывода и обеспечивающие обслуживание новых устройств или нестандартное использование имеющихся устройств. Драйверы загружаются в оперативное запоминающее устройство при загрузке ОС, а их имена указываются в файле конфигурации CONFIG.SYS.

Загрузка MS DOS в оперативную память с диска

ОС хранится во внешней памяти на жестком или (реже) на гибком диске. Для работы ПК необходимо, чтобы основные модули ОС находились в оперативной памяти. Поэтому после включения ПК организована автоматическая перезапись (загрузка) ОС с диска в оперативную память.

Запуск ПК и подготовка ОС к работе включает следующие шаги:

1. При включении ПК управление передается базовой системе ввода-вывода BIOS. BIOS выполняет тестирование памяти, проверку состояния аппаратуры и инициализирует устройства. Параметры конфигурации ПК извлекаются из так называемой энергонезависимой памяти. При нажатии клавиши перед инициализацией устройств можно передать управление программе изменения параметров конфигурации.

2. Управление конфигурацией ПК (задание параметров жесткого диска, указание системного диска, задание пароля) выполняется с помощью программы Setup.

3. Вызов загрузчика (BOOT RECORD) и загрузка с его помощью в оперативную память модуля расширения IO.SYS и базового модуля MSDOS.SYS.

4. Загрузка командного процессора COMMAND.COM.

5. Обработка файла конфигурации CONFIG.SYS, содержащего команды подключения необходимых драйверов.

6. Обработка командного файла AUTOEXEC.BAT. С помощью этого файла можно произвести настройку параметров ОС. Например, создать виртуальный диск, обеспечить смену режимов печати, загрузить вспомогательные программы и т.д.

Технология работы в MS DOS. Общие сведения о командах

Работа в ОС MS DOS организуется командами. Они вызывают определенное действие: организуют передачу информации, вырабатывают необходимый управляющий сигнал, подключают внешнее устройство для организации процесса ввода-вывода информации и т.д.

Команда технически реализована программой в машинных кодах и хранится либо в файле на диске (внешняя команда), либо входит в состав командного процессора COMMAND.COM (внутренняя команда). По порядку запуска внутренние и внешние команды не различаются. При запуске внешних команд необходимо удостовериться, что файлы, в которых они находятся, существуют на диске и находятся на "видимой" (компьютеру) директории. Как и любая другая программа, команда имеет уникальное имя и всегда имеет тип COM или EXE.

Ввод команды осуществляется в командной строке в соответствии с определенными правилами, заданными в виде формата.

Командная строка – строка экрана дисплея, начинающаяся с приглашения ОС. Командная строка состоит из информации подсказки, указателя ввода и курсора. Обычная информация подсказки указывает на диск и директорию, где в это время производится работа.

Формат команды – правило формирования команды пользователем с клавиатуры.

При формировании команды в соответствии с установленным форматом необходимо соблюдать следующие правила:

1) формат команды состоит из имени команды (латинскими буквами без указания типа) и отделенных от него одним пробелом параметров, уточняющих действие команды;

2) в большинстве случаев параметры между собой пробелом не разделяются, а в качестве разделителя часто используется символ '/';

3) параметрами могут быть: имя логического диска, путь, имя файла, тип файла, латинские буквы, символы, цифры;

4) параметры в формате могут и отсутствовать, что указывается с помощью квадратных скобок '[' и ']'.
Обобщенный формат команды можно представить в следующем виде:

<имя команды> [<параметры>].

Пример: C:\>DIR D:\USER*.TXT/P

Здесь:

C:\> – приглашение ОС MS DOS;

DIR – имя команды;

D:\USER*.TXT/P – параметры.

Эта команда вызывает с помощью параметра D:\USER*.TXT/P на экран записи обо всех файлах типа 'TXT' из директории первого уровня 'USER' логического диска 'D'. Вызов записей производится постранично, на что указывает параметр '/P'.

Процедура ввода команды состоит в следующем:

1) в соответствии с форматом в командной строке набирают имя команды и необходимые параметры;

2) нажимают клавишу ввода, что служит сигналом начала анализа структуры набранной команды. При отсутствии ошибок в формате команды она будет выполнена, иначе на экран выдается сообщение: Bad command or filename (Неверная команда или имя файла);

3) при невыполнении команды просматривают вводимую конструкцию и вновь вводят ее, но уже в откорректированном варианте.

Порядок действий при выполнении команды MS DOS

После ввода команды с клавиатуры MS DOS выполняет следующие действия:

1. MS DOS анализирует первое слово командной строки (последовательность символов до первого пробела) с целью выяснить – задано ли просто имя, неполный адрес или точный адрес. Основным ключ к анализу – наличие символов ':' и '\'.

2. Если задано просто имя, MS DOS ищет его сначала в файле COMMAND.COM, затем в текущей директории, затем в директориях, перечисленных в команде PATH, записанной в файле авто-запуска AUTOEXEC.BAT. Если расширение в имени опущено, поиск ведется по собственному имени с подстановкой расширения в следующем порядке: COM, EXE, BAT.

3. Если задан неполный адрес, MS DOS ищет программу либо на текущем диске, либо начиная с текущей директории. Например, если указано: \SIMP\REM.EXE, то поиск файла REM.EXE ведется в каталоге SIMP текущего диска. Если указано: C\SIMP\REM.EXE, то MS DOS ищет файл REM.EXE в поддиректории 'C' текущей директории.

4. Если задан точный адрес, то MS DOS просто следует по указанному пути, не обращая внимания ни на текущий каталог, ни на директории, перечисленные в команде PATH.

5. Найдя программу, MS DOS загружает ее и передает ей в качестве параметров все, что набрано в командной строке.

6. После завершения программы на экран вновь выводится приглашение MS DOS.

7. Если программа не найдена, на дисплей поступает сообщение: Bad command or filename (имя команды или файла указано неверно), и выдается приглашение MS DOS.

Команды MS DOS общего назначения

По мере необходимости пользователь может использовать следующие команды, называемые командами общего назначения:

1. CLS – очистка экрана от выведенной до этого информации.
2. ECHO <сообщение> – печать сообщения на экране. Команды ECHO OFF и ECHO ON соответственно запрещают и разрешают печать сообщения других команд.
3. DATE – вывод на экран или установка текущей даты в формате "мм-дд-гг".
4. TIME – вывод на экран или установка системного времени в формате "чч:мм".
5. PROMPT \$<тип информации>\$<вид указателя> – определение системной подсказки. Тип информации задается символами: D – текущая дата, P – текущий диск и путь, N – только текущий диск, T – текущее время. Вид указателя задается символами: G (на экране появится символ '>') или L (на экране появится символ '<'). Обычно эту команду используют в следующем виде: PROMPT \$P\$G.
6. VER – вывод на экран номера версии ОС на этом ПК.

Основные команды для работы с директориями.

1. Команда DIR – просмотр директории

Работа на ПК, как правило, начинается с просмотра директории, например, чтобы убедиться в том, что нужный вам файл существует. Часто необходимо просмотреть содержимое пассивной директории.

В зависимости от параметров, допустимых в структуре команды, можно просмотреть записи директории в стандартной или усеченной форме с выводом только полных имен файлов, а также при большом содержании директории выводить ее постранично.

Формат команды:

DIR [Имя дисководов:][Путь\][Имя файла][Параметры]

Если имя дисководов и/или путь отсутствуют в команде, то подразумевается текущий дисковод и текущая директория. Параметры (ключи) задают порядок вывода списка файлов и директорий.

Назначение основных параметров (ключей):

/P – постраничный вывод содержимого директории на экран. Для продолжения вывода следует нажать любую клавишу;

/W – вывод только полных имен файлов и директорий;

/A – индикация содержимого директорий с атрибутами;

/O – задание порядка сортировки выводимых сведений.

Примеры:

C:\>DIR Вывод содержимого корневой текущей директории на экран

C:\USER1>DIR *.BAK Вывод на экран всех имен файлов типа BAK из текущей директории первого уровня USER1

C:\>DIR A: Вывод на экран содержимого пассивного дискового A

C:\B1>DIR B2 Вывод на экран содержимого пассивной директории второго уровня B2, находящейся в директории первого уровня B1

C:\USER1>DIR \B1\B2/P Вывод на экран постранично содержимого пассивной директории B2. Эта директория находится в другой ветке иерархической структуры директории, чем текущая директория USER1

C:\B1>DIR /W Вывод на экран записей текущей директории B1 в усеченном формате (только полные имена файлов и директорий)

2. Команда MD – создание директории

Новую директорию можно создать командой MD в текущей директории или, если указан путь, в пассивной директории.

Формат команды:

MD [Имя дискового:][Путь\]Имя директории

Примеры:

C:\>MD USER1 Создание (в текущей корневой директории) директории первого уровня USER1

C:\>MD USER1\USER2 Создание (в директории первого уровня USER1) директории второго уровня USER2

C:\T1\T2>MD USER1\USER2 Создание директории второго уровня USER2, если ОС находится в другом каталоге второго уровня T2.

3. Команда RD – уничтожение директории

Эта команда уничтожает только пустую директорию. Предварительно необходимо удалить из нее командой DEL все файлы, а затем командой DIR убедиться в том, что она пустая.

Формат команды:

RD [Имя дискового:][Путь\]Имя директории

Примеры:

C:\>RD USER1 Удаление директории USER1 из корневой директории

C:\B1>RD \USER1\USER2 Удаление пассивной директории второго уровня USER2, если ОС находится в текущей директории B1

4. Команда CD – переход в другую директорию

Иногда необходимо перейти в другую директорию и сделать ее текущей. В этом случае следует воспользоваться командой CD.

Формат команды:

CD [Имя дискового:][Путь\]Имя директории

Для перехода в родительскую директорию достаточно вместо имени директории задать '..' (две точки). Для перехода в корневую директорию задают символ '\'.
Примеры:

C:\>CD USER1 Переход в директорию USER1 из корневой директории. После ввода команды приглашение примет вид: C:\USER1>

C:\B1\B2>CD \USER1 Переход из директории второго уровня B2 в директорию первого уровня USER1, находящуюся в другой ветви иерархической структуры. После ввода команды приглашение примет вид: C:\USER1>

A:\>CD C:\USER1 Переход из корневой директории диска A в директорию первого уровня диска C. После ввода команды приглашение примет вид: C:\USER1>

C:\M1\M2\M3>CD .. Переход в родительскую директорию. После ввода команды приглашение примет вид: C:\M1\M2>

C:\F1\F2>CD \ Переход в корневую директорию. После ввода команды приглашение примет вид: C:\>

Основные команды для работы с файлами.

1. Команда TYPE – просмотр текстового файла.

Командой TYPE удобно пользоваться для просмотра содержимого текстового файла на экране дисплея или на принтере. После запуска команды текст (содержимое файла) выводится непрерывным потоком, причем скорость смены кадров с текстом на экране настолько велика, что прочесть его практически невозможно. Для приостановки вывода текста нажимают одновременно две клавиши: <CTRL> и <S>. Нажатие затем любой клавиши возобновит вывод текста.

Формат команды для вывода на экран:

TYPE [Имя дисковода:][Путь\]Полное имя файла

Формат команды для печати на принтере:

TYPE [Имя дисковода:][Путь\]Полное имя файла>PRN

Примеры:

C:\>TYPE ROK.TXT Вывод на экран содержимого файла ROK.TXT, расположенного в корневой директории активного дисковода

C:\>TYPE A:\RED\LOT.TXT Вывод на экран содержимого файла LOT.TXT, расположенного в директории RED первого уровня пассивного дисковода A

C:\>TYPE \B1\BOOK.TXT>PRN Печать на принтере содержимого файла BOOK.TXT, расположенного в директории B1 первого уровня текущего дисковода C

2. Команда DEL – удаление файлов

Можно удалять как один файл, так и группу файлов, указывая для группы в шаблоне имени файла символы '*' или '?'.
Формат команды:

DEL [Имя дисковода:][Путь\]Полное имя файла>[/P]

Параметр [/P] служит для вывода на экран запроса на подтверждение

удаления.

При вводе команды удаления всех файлов (DEL *.*) ОС задает вопрос Are You sure (Y/N)? (Вы уверены?). Если вы не передумали, нажмите клавишу <Y>, в противном случае – клавишу <N>.

Примеры:

C:\>DEL TOST.TXT Удаление файла TOST.TXT из корне-вой директории текущего дисковогода C

C:\>DEL A:\AR\B.TXT Удаление файла B.TXT из директории первого уровня AR пассивного дисковогода A

C:\>DEL \A1\A2*.BAS Удаление всех файлов типа BAS из директории второго уровня A2 текущего дисковогода C

C:\F1>DEL F2*.* /P Удаление с подтверждением всех файлов из директории второго уровня F2 текущего дисковогода C, подчиненной текущей директории F1

3. Команда COPY – копирование файлов

Команда используется для создания копий существующих файлов, вывода содержимого файла на внешнее устройство, объединения содержимого нескольких файлов.

Команда COPY допускает выполнение функций над группой файлов, и тогда в шаблоне имени файла используются символы '*' или '?'.

Формат команды для копирования файлов:

COPY [Имя дисковогода-источника:][Путь\]Полное имя файла-источника[Имя дисковогода-приемника:][Путь\][Полное имя файла-приемника][N]

Обязательным параметром является только полное имя файла-источника. Если копируется файл (группа файлов) с тем же именем (именами), то достаточно указать только полное имя файла-источника. Имя дисковогода и путь нужно указывать при работе с пассивным дисководом и директорией.

Примеры:

C:\>COPY ROK.PAS A: Копирование файла ROK.PAS из корневой директории текущего дисково-да C на диск A с тем же именем

C:\>COPY A1\A2\R.TXT \B1\B2\B3 Копирование файла R.TXT из директории A2 второго уровня в директорию третьего уровня B3 с тем же именем

C:\>COPY A:ROST.BAS Копирование файла ROST.BAS с пассивного дисковогода A в корневую директорию текущего диско-вода C с тем же именем

C:\>COPY A:T.TXT A1\A2\S.TXT Копирование файла T.TXT с диска пассивного дисковогода в директорию второго уровня A2 текущего дисковогода. Полученной копии файла присваивается новое имя S.TXT

C:\A1>COPY *.BAS B:/V Копирование всех файлов типа BAS из текущей директории первого уровня A1 дисковогода C на диск пассивного дисковогода B с одно-временным контролем процесса копирования

Формат команды объединения нескольких файлов:

COPY [Имя дисковод:][Путь\]Полное имя файла + [Имя дисковод:][Путь\]Полное имя файла + ... [Имя дисковод:][Путь\]Полное имя файла-приемника

Имена объединяемых файлов перечисляются в команде COPY через знак '+'.
Имя результирующего файла записывается последним и отделяется от имен объединяемых файлов пробелом.

Содержимое результирующего файла представляет собой подсоединенное друг за другом содержимое исходных файлов в соответствии с порядком следования их имен.

Примеры:

C:\>COPY M1.TXT+M2.TXT \K1\SUM.TXT Объединение двух текстовых файлов M1.TXT и M2.TXT в файл SUM.TXT, который будет записан в директорию первого уровня K1

C:\>COPY T1.TXT+T2.TXT Объединение двух текстовых файлов T1.TXT и T2.TXT. К содержимому файла T1.TXT добавляется содержимое файла T2.TXT, и результат объединения будет храниться в файле с именем T1.TXT

Форматы команд для обмена данными между внешним устройством и файлом, хранящимся на диске:

COPY Имя внешнего устройства (откуда) [Имя дисковод:][Путь\]Полное имя файла (куда)

COPY [Имя дисковод:][Путь\]Полное имя файла (откуда) Имя внешнего устройства (куда)

COPY Имя внешнего устройства (откуда) Имя внешнего устройства (куда)

Под внешним устройством здесь понимается любое устройство, кроме системного блока и дисководов. В ОС приняты соглашения относительно имен внешних устройств. К наиболее употребительным относятся:

- CON – клавиатура и дисплей (консоль);
- PRN или LPT1 – основной принтер.

Примеры:

C:\>COPY T.TXT PRN Печать содержимого текстового файла на принтере

C:\>COPY CON S.TXT Заполнение файла S.TXT поступающими с клавиатуры символами

C:\>COPY CON PRN Все символы, набираемые с клавиатуры, печатаются, минуя центральную часть компьютера, т.е. компьютер используется как пишущая машинка. Одновременно компьютер может обрабатывать информацию в соответствии с программой, где не требуется обращение к принтеру

Порядок выполнения работы

1. Ознакомиться с основными теоретическими положениями.
2. Ответить на контрольные вопросы.
3. Получить у преподавателя вариант задания.
4. Выполнить задание, результат показать преподавателю.

Контрольные вопросы

1. Что такое файл, характеристики файла?
2. Понятие имени файла и полного имени файла.
3. Каковы правила при задании имени файла в MS DOS?
4. Какие символы используются в шаблоне имени файла?
5. Доступ и три способа организации доступа к файлу.
6. Функции и назначение базовой системы ввода-вывода.
7. Функции и назначение базового модуля.
8. Функции и назначение командного процессора.
9. Утилиты, внешние команды и драйверы.
10. Команды MS DOS общего назначения.
11. Основные команды для работы с директориями.
12. Основные команды для работы с файлами.

Практическое занятие №3

Работа с базами данных

Цель работы: освоение приемов создания и редактирования документов Access.

Общие сведения о реляционных базах данных

Базой данных (БД) называется упорядоченный набор сведений о каком-то множестве объектов (людей, книг, организаций и т.п.). Наиболее распространены реляционные БД с табличной организацией данных. Для работы с БД применяются специальные программы – системы управления базами данных (СУБД). В данной работе рассматривается СУБД Access 2007.

Основой БД в Access являются связанные между собой

таблицы, которые служат хранилищами данных⁸. Каждая строка таблицы содержит одну запись – набор сведений об одном объекте. Записи обо всех объектах в таблице имеют одинаковую структуру и состоят из отдельных полей, содержащих для разных объектов значения одного и того же атрибута(признака). Значения одного поля для разных объектов хранятся в одном столбце таблицы.

Структура таблиц БД должна подчиняться определенным правилам, определяющим нормальные формы таблиц.

Первая нормальная форма (1НФ) требует, чтобы значения полей в записях были атомарными (неделимыми). Например, фамилия, имя и отчество человека должны рассматриваться как три отдельных атрибута, т.е. помещаться в отдельных полях.

Вторая нормальная форма (2НФ) требует, чтобы:

- 1) таблица находилась в 1НФ;
- 2) в таблице имелся первичный ключ – поле, значения которого однозначно определяли бы любую запись.

Каждой записи в таблице должно соответствовать какое-то значение ключа (ключевое поле не может быть пустым), кроме того, разные записи не могут иметь одно и то же значение ключа. Если известно значение ключа, то по нему можно безошибочно найти в таблице нужную запись и определить значения всех неключевых полей этой записи (говорят, что неключевые поля связаны функциональной зависимостью с ключевым).

Третья нормальная форма (3НФ) требует, чтобы:

- 1) таблица находилась в 2НФ;
- 2) никакие два неключевых поля не были связаны друг с другом функциональной зависимостью (значение одного не-ключевого поля не должно влиять на значение другого).

Анализ требований остальных НФ выходит за рамки данной работы. Требования НФ носят рекомендательный характер. При их нарушении БД

может функционировать, но работа с ней усложняется и возрастает вероятность появления ошибок, вызванных влиянием человеческого фактора.

Стремление к соблюдению требований 3НФ приводит к тому, что все данные оказывается невозможно собрать в одну таблицу – в нее могут попасть функционально связанные не-ключевые атрибуты. Поэтому данные приходится размещать в нескольких таблицах, связь между которыми устанавливается с помощью ключевых полей.

Если в двух таблицах имеется одинаковое ключевое поле, то каждой записи одной таблицы соответствует одна запись другой таблицы, имеющая такое же значение ключа. Такая связь таблиц называется «один к одному».

Другой тип соединения получается, если какое-то поле является ключевым в одной таблице, и неключевым в другой. Тогда во второй таблице может существовать несколько записей с одинаковым значением этого поля, и все они будут связаны с одной и той же записью первой таблицы. Эта связь называется «один ко многим». Неключевое поле второй таблицы, посредством которого осуществляется связь, называется ее внешним ключом.

Общая структура всех таблиц БД и их связей называется схемой данных базы.

Документы базы данных

При работе с БД могут создаваться три типа документов:

1. Форма – служит для ввода, просмотра, корректировки и удаления записей. В отличие от выполнения тех же операций непосредственно в таблице, форма обычно показывает пользователю только одну запись. Представление записи может иметь вид готового бланка какого-то документа (анкеты, ведомости и т.п.) или содержать стандартные управляющие элементы интерфейса приложений Windows (окна ввода, кнопки, флажки и т.п.). Форма может также выборочно показывать пользователю только отдельные поля записи, подлежащие исправлению.

2. Запрос – это инструмент, который служит для выбора и изменения данных. В СУБД Access существуют различные виды запросов, например:

а) запрос на выборку позволяет выбрать из таблицы или их совокупности отдельные записи или их фрагменты по какому-то признаку;

б) запрос на изменение служит для автоматического изменения содержимого отдельных полей в выбранных записях;

в) запрос на создание таблицы позволяет сохранить выбранные записи или их фрагменты в отдельной таблице.

Запросы служат основой для создания других типов документов – форм и отчетов.

3. Отчет – представление данных, полученных на основе запроса в виде оформленного и подготовленного для печати документа (справки, ведомости, приказы и др.).

Access предлагает два режима создания документов: режим Мастера и режим Конструктора. При создании документов в режиме Мастера значительную часть работы по созданию документа берет на себя компьютер, предлагая пользователю готовые варианты. В режиме Конструктора определение структуры и оформление документа осуществляются пользователем.

При создании таблиц данных используются режим таблицы и режим Конструктора.

Заполнение таблицы может производиться:

- 1) вручную (в режиме таблицы или с помощью форм).
- 2) путем импорта данных из другой БД.
- 3) путем импорта данных из таблицы Excel.

База данных хранится на диске в виде единого файла, имеющего расширение .accdb. Число записей в таблицах ограничивается только емкостью диска. Максимальное число полей отдельной таблицы может достигать 255.

Типы данных в Access

В СУБД Access предусмотрены следующие типы полей:

- Числовой. Числовые поля могут иметь разные форматы – целочисленные, вещественные, денежные и др., аналогично форматам чисел в электронных таблицах Excel.

- Текстовый (символьный) – текстовая строка. Число символов (длина строки) задается при создании таблицы и не может быть более 255.

- Дата/время.

- Логический тип (да/нет или true/false).

- «Мемо» (примечание) – текст большой длины (до 65 536 символов). Мемо-поля хранятся отдельно от таблицы, а в само поле стандартной длины заносится адрес 1-го байта сохраняемого текста (указатель).

- Счетчик – поле автоматической нумерации записей. В Access это поле нужно создавать принудительно (хотя потом оно управляется автоматически). Значение счетчика увеличивается на единицу для каждой новой записи и закрепляется за записью. При удалении записи соответствующее значение счетчика тоже удаляется и не присваивается никакой другой записи. Обычно поле счетчика является первичным ключом таблицы.

- OLE – внедренный объект, созданный в другой программе (рисунки, видео- и аудиофайлы; в этом случае в поле таблицы также заносится только указатель на 1-й байт файла объекта).

- Гиперссылка на объект интернета.

- Подстановка – набор стандартных значений любого вышеперечисленного типа (кроме счетчика), выбираемых из готового

списка. Подстановки используют, если возможные значения поля образуют относительно небольшое дискретное множество. Применение подстановок позволяет уменьшить вероятность случайных опечаток при заполнении таблицы.

Проектирование базы данных

В предлагаемых далее практических работах будет создана БД, моделирующая электронный каталог библиотеки вуза. Рассмотрим проектирование этой БД.

Разработка БД начинается с определения перечня различных множеств объектов, сведения о которых будут храниться в БД. Для каждого множества определяется набор атрибутов, описываемый базой. Если возможно, для атрибута определяется множество допустимых значений. Оно называется домом атрибута, а условия, его определяющие – ограничения домена.

Определение домена атрибута позволяет при вводе и корректировке данных контролировать вводимые значения, снижая вероятность случайных опечаток.

Затем составляется схема данных. После этого можно приступить к созданию и заполнению таблиц.

В простейшем случае наша БД должна описывать два множества объектов и связи между ними – множество книг, хранящихся в библиотеке, и множество студентов-читателей.

Рассмотрим множество книг. Естественные атрибуты любой книги – Ф.И.О. авторов, название книги, издательство, год издания, количество страниц, дата поступления в библиотеку. Учтем также цену книги на момент поступления в библиотеку (для определения размера компенсации при утере). Ни одно из этих полей не может однозначно определить запись о книге (т.к. в библиотеку может одновременно поступить несколько одинаковых экземпляров книги), поэтому для различения записей введем дополнительное ключевое поле типа «счетчик» (значение счетчика одновременно будет определять инвентарный номер книги).

Кроме того, для облегчения поиска литературы по определенным читательским запросам следует учитывать тематику книг. В бумажных каталогах (картотеках) библиотек тематика учитывается в отдельном тематическом каталоге, который составляется по иерархическому (уровневому) принципу, например, так, как показано в табл. 3.1

В нашей БД мы будем учитывать только эти два уровня иерархии. Верхний (более общий) уровень назовем «Раздел», нижний (детальный) – «Тема». Если указать в таблице тему книги, то это автоматически определит раздел, к которому она относится, так что эти атрибуты функционально связаны. Т.к. поля этих атрибутов не будут ключевыми в таблице «Список книг», они не могут находиться в одной таблице, ибо это нарушит требования 3НФ. Для хранения данных об иерархии разделов и тем необходимо создать отдельную таблицу «Тематика», включающую только два поля – «Тема»

(ключевое) и «Раздел». В таблицу «Список книг» включается поле «Тема», которое будет внешним ключом для связывания этих двух таблиц.

Таблица 3.1

Раздел	Тема
Математика	Математический анализ
	Прикладная математика
Информатика	Программирование
	Офисные программы
	Базы данных
Физика	Механика
	Теплофизика
Естественные науки	Химия
	Биология
Общественные науки	История
	Философия
Технические науки	Машиностроение
	Химическая и пищевая
	Строительство

Наконец, в таблице «Список книг» надо отразить данные о том, какому читателю выдана книга. Для этого необходим список читателей, для которого создадим соответствующую таблицу.

Необходимый набор данных о читателях-студентах включает фамилию студента, код учебной группы и название факультета, № студенческого/читательского билета (это поле является первичным ключом таблицы «Список читателей» и в нашей учебной БД тоже будет введено как счетчик записей). При этом код группы и название факультета также функционально связаны (код группы определяет факультет) и включение их в одну таблицу нарушит требования 3НФ. Поэтому надо ввести отдельную таблицу «Учебные группы» с записями из двух полей: «Группа» (ключевое) и «Факультет». В таблицу «Список читателей» включается только поле «Группа», которое будет внешним ключом для связи с таблицей «Учебные группы».

В таблицу «Список книг» в качестве внешнего ключа для связи с таблицей «Список читателей» вводится поле «Читатель», в котором будет записываться номер читательского билета студента, которому выдана книга.

Для описания ситуации, когда книга никому не выдана, в таблицу «Список читателей» введем фиктивного читателя – «Хранилище».

Итак, наша база должна включать 4 таблицы: «Список книг», «Тематика», «Список читателей», «Учебные группы». Их описание приведено ниже в табл. 3.2–3.5

Таблица 3.2

Структура таблицы «Список книг»

Имя поля	Содержание поля	Тип поля и ограничения домена
№	Инвентарный номер	Счетчик (ключевое поле)
Авторы	Авторы книги (в случае большого количества авторов указывается	Текстовый, длина 100 символов
Название	Название книги	Текстовый, длина 200
Место издания	Город, в котором издана книга (для крупнейших городов сокращенно: М., СПб.) и название	Текстовый, длина 50
Год	Год издания	Числовой, формат – фиксированный, число десятичных знаков – 0 (т.е. целое число); значение не может превышать текущий год на момент заноса в таблицу
Страницы	Количество страниц	Числовой (целое число)
Цена	Цена книги	Числовой, формат – денежный
Поступила	Дата регистрации книги в каталоге	Дата, формат – краткий формат даты; значение не может превышать текущую дату на момент заноса в таблицу
Тема	Основная тематика книги	Текстовый, длина 50; значения должны принадлежать к множеству значений поля «Тема» из таблицы «Тематика»
Читатель	№ читательского билета	Числовой (целый); значения берутся из списка номеров читательских билетов в таблице «Список читателей»

Таблица 3.3

Структура таблицы «Тематика»

Имя поля	Содержание поля	Тип поля и ограничения домена
Тема	Тема книги (ключевое поле)	Текстовый, длина 50
Раздел	Тематический раздел книги	Текстовый, длина 50; значения берутся из списка разделов, приведенного выше в табл. 4.1

Таблица 3.4

Структура таблицы «Список читателей»

Имя поля	Содержание поля	Тип поля и ограничения домена
№ билета	№ читательского билета	Счетчик (ключевое поле)
Фамилия	Фамилия	Текстовый, длина 25
Группа	Шифр учебной группы	Текстовый, длина 10; значения должны принадлежать к множеству значений поля «Группа» из таблицы «Учебные группы»

Таблица 3.5

Структура таблицы «Учебные группы»

Имя поля	Содержание поля	Тип поля и ограничения домена
Группа	Шифр учебной группы (ключевое поле)	Текстовый, длина 10
Факультет	Наименование факультета	Текстовый, длина 20; может иметь одно из трёх значений: «Механический», «Технологический», «Экономический»

Задания для самостоятельной работы. Часть I.

1. Создать БД, сохранив ее в собственной папке, размещенной в папке «Мои документы».

2. Создать таблицу «Тематика» со структурой, показанной в таблице 3.3. Создать список подстановок для поля «Раздел». Заполнить таблицу данными в соответствии с табл. 3.1.

3. Создать таблицу «Учебные группы» со структурой, показанной в табл. 3.5. Создать список подстановок для поля «Факультет». Заполнить таблицу, указав в ней шифры шести различных групп трех разных факультетов.

4. Создать таблицу «Список читателей» со структурой, показанной в табл. 3.4. Для поля «Группа» создать список подстановок на основе поля «Группа» из таблицы «Учебные группы». Заполнить таблицу фамилиями не менее чем десяти читателей путем импорта данных из таблицы Excel. Указать для читателей группы, соответствующие трем разным факультетам.

5. Создать таблицу «Список книг» со структурой, показанной в табл. 3.2. Для поля «Тема» создать список подстановок на основе поля «Тема» из таблицы «Тематика». Для поля «Читатель» создать список подстановок на основе поля «№ билета» из таблицы «Список читателей». Заполнить таблицу данными о книгах из табл. 3.6 согласно индивидуальному варианту (табл. 3.7). Номера читательских билетов указать произвольно, используя созданный список подстановок.

Порядок выполнения работы Создание базы данных

1. Запустить СУБД Access, на экране возникает окно приложения, показанное на рис. 3.1.

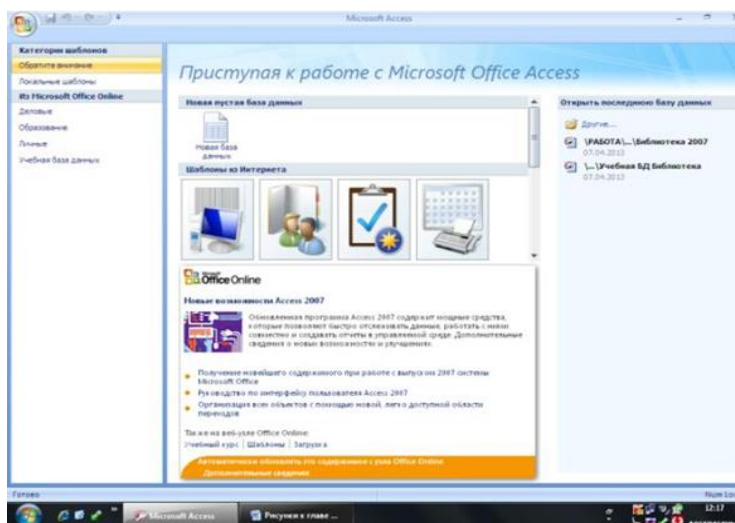


Рис. 3.1. Создание базы данных

Справа в окне можно видеть список БД, с которыми производилась работа в последнее время. В центральной части окна находится предложение пользователю создать новую БД.

2. Щелкнуть по кнопке [Новая база данных]. Справа появится окно ввода «Имя файла» для указания имени создаваемой БД. Под окном показан предлагаемый по умолчанию маршрут для файла БД.

3. Щелкнуть кнопку с изображением открытой папки справа от окна ввода «Имя файла». Она служит для выбора расположения создаваемой БД. После щелчка открывается обычное окно сохранения документа, в котором необходимо указать имя БД и выбрать для ее размещения собственную папку, затем щелкнуть клавишу [Создать].

После создания БД в окне документа появляются две панели. Левая называется областью переходов. В ней отображаются заголовки элементов БД – таблиц, форм, запросов и др. Область переходов можно свернуть, щелкнув кнопку [<<] в ее заголовке; также можно перетаскивать ее границу мышью. Для открытия доступа ко всем элементам БД надо щелкнуть значок списка в заголовке области переходов и выбрать пункт «Все объекты Access».

Правая панель отображает элемент БД, с которым производится работа («активный»). Если открыть сразу несколько документов БД, в верхней части панели появятся закладки этих документов.

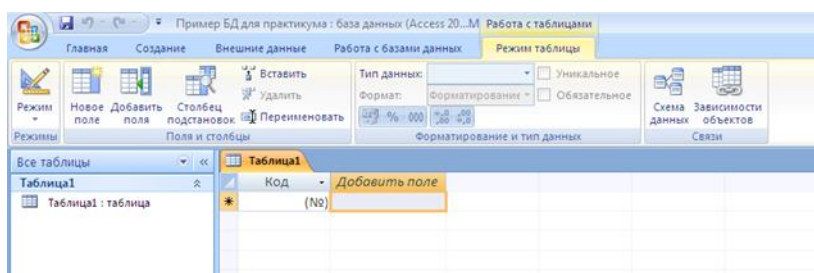


Рис. 3.2. Начало создания таблицы

В начале работы по созданию БД в правой части возникает начальный шаблон первой таблицы (рис. 3.2). Он отображает режим таблицы. В этом режиме можно создавать поля, устанавливать типы данных, заполнять записи данными.

Однако создавать таблицы мы будем в режиме Конструктора, а режим таблицы использовать для их заполнения.

Создание и заполнение таблицы «Тематика»

4. Для перехода в режим Конструктора щелкнуть заголовок «Таблица 1» правой кнопкой мыши и в контекстном меню выбрать «Конструктор». В появившемся окне сохранения указать имя создаваемой таблицы

(«Тематика») и сохранить таблицу. Окно Access принимает вид, показанный на рис. 3.3.

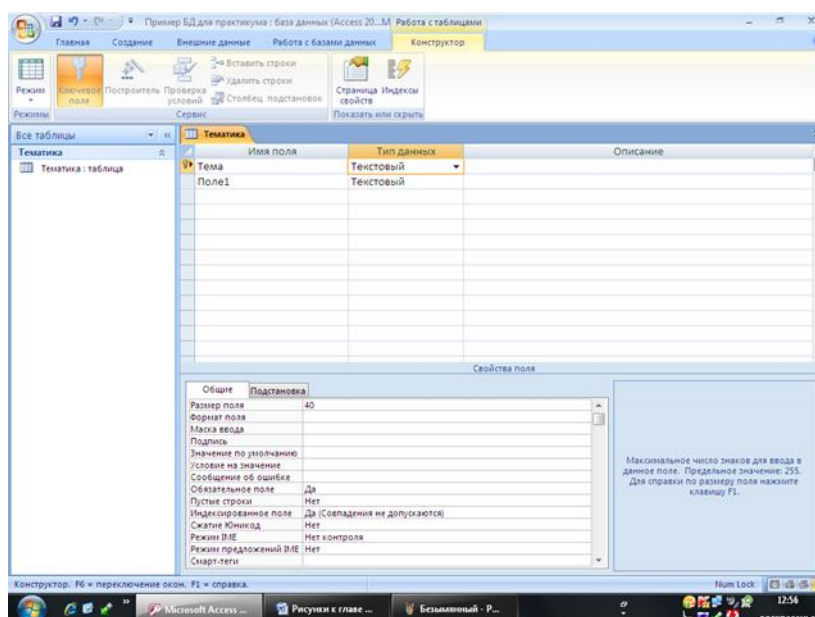


Рис. 3.3. Конструктор таблиц

В области переходов перечислены имеющиеся в БД таблицы (пока имеется только таблица «Тематика»). Не выходя из режима Конструктора, можно открыть любую таблицу и изменить ее структуру. Справа виден перечень полей открытой таблицы. Первое поле по умолчанию предполагается ключевым (об этом говорит символ ключа слева от имени поля). Если поле не должно быть ключевым, надо установить курсор в ячейку с его именем и щелкнуть на вкладке «Конструктор» клавишу [Ключевое поле] – символ ключа исчезает. Таким же образом можно при необходимости назначить поле ключевым.

5. Оставить первое поле ключевым, а в окне имени поля ввести имя «Тема». В окне «Тип данных» из предлагаемого списка типов выбрать «Текстовый».

В нижней части окна Конструктора приводится список свойств создаваемого поля. Для поля «Тема» назначить размер 50. Из прочих свойств назначить (вариант выбирается из предлагаемого списка): «Обязательное поле» – «Да», «Пустые строки» – «Нет», «Индексированное поле» – «Да». Это соответствует требованиям к ключевому полю (отсутствие пустых и повторяющихся значений).

6. Затем, заполняя соответствующие ячейки списка имен полей и их свойств, создать поле «Раздел» (текстовое, размер 50, прочие свойства – по умолчанию). Для будущего ввода данных в это поле надо создать для него список подстановок.

7. Для быстрого создания списка в разделе свойств можно просто щелкнуть заголовок вкладки «Подстановка» и последовательно выбрать:

«Тип элемента управления» – «Поле со списком», «Тип источника строк» – «Список значений», «Источник строк» – щелкнуть возникшую в ячейке кнопку с многоточием.

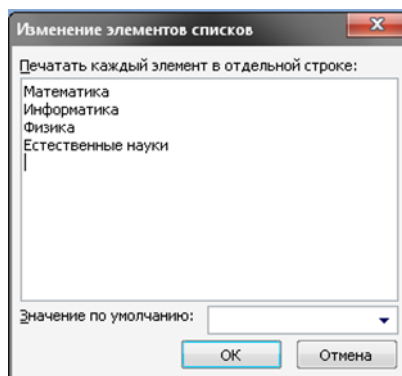


Рис. 3.4. Создание списка подстановок

В появившееся после этого окно ввода (рис. 3.4) ввести названия разделов из таблицы 3.1 (после каждого названия нажимать клавишу ввода). После ввода всех названий проверить список и устранить опечатки, затем щелкнуть [ОК].

8. После создания таблицы сохранить ее, щелкнув клавишу [Сохранить] на панели быстрого доступа.

9. Для заполнения таблицы перейти в режим таблицы. Для этого дважды щелкнуть имя нужной таблицы в области переходов.

10. Заполнить таблицу в соответствии с заданием.

Заполнение таблицы производится аналогично заполнению таблиц Word или Excel. Как только начинается заполнение очередной строки (записи), внизу автоматически появляется пустая строка для следующей записи. При необходимости можно мышью передвигать границы столбцов, меняя их ширину.

Значения поля «Тема» вводятся с клавиатуры. Для заполнения значений поля «Раздел» используется созданный ранее список значений, раскрываемый щелчком по значку списка, который появляется в ячейке, когда в нее устанавливается курсор.

11. Сохранить заполненную таблицу.

Создание и заполнение таблицы «Учебные группы»

12. Для создания новой таблицы «Учебные группы» выполнить команду «Создание» – «Таблицы» – [Конструктор таблиц]. В окне Конструктора (рис. 3.3) рядом с закладкой таблицы «Тематика» появится закладка «Таблица 1» с пустой структурой таблицы. На ее основе создать, сохранить и заполнить таблицу с указанными в задании полями и их свойствами. Назначить поле «Группа» ключевым. Процесс создания структуры, назначения свойств полей, создания списка подстановок для поля «Факультет» и заполнения таблицы ничем не отличается от описанного выше

для таблицы «Тематика». При заполнении таблицы ввести «пустую» группу с шифром «-» (прочерк). Ей в БД будет соответствовать фиктивный читатель «Хранилище». В поле «Факультет» для хранилища также поставить прочерк.

После заполнения таблицы сохранить ее.

Создание таблицы «Список читателей»

13. Создать в режиме Конструктора таблицу «Список читателей». Назначить поле «№» ключевым. Задать свойства полей «№» и «Фамилия».

14. При установлении типа данных для поля «Группа» выбрать из списка: «Мастер подстановок». Запускается процесс создания подстановки, в ходе которого последовательно открываются диалоговые окна для настройки параметров списка. Для перехода к следующему шагу процесса надо щелкнуть в окне клавишу [Далее].

а) в первом окне установить переключатель на позицию «Объект будет использовать значения из таблицы или запроса»;

б) в следующем окне выбрать из списка имеющихся таблиц ту, в которой содержатся элементы списка для подстановок в поле «Группа», т.е. таблицу «Учебные группы»;

в) в третьем окне пометить слева щелчком поле, содержащее элементы списка – «Группа» – и щелчком по кнопке [>] внести его в список выбранных полей (другая кнопка – [>>] – позволяет вносить в список сразу все имеющиеся поля);

г) в следующем окне установить порядок сортировки элементов списка подстановок – «По возрастанию»;

д) в следующих окнах выбрать желаемую ширину поля таблицы для размещения значений списка и заголовков столбца, содержащего элементы списка; использовать варианты, предлагаемые по умолчанию. В последнем окне щелкнуть [Готово].

15. Сохранить таблицу под именем «Список читателей».

16. Провести заполнение столбца «Фамилия» путем импорта данных из Excel:

а) запустить Excel. Заполнить один из столбцов фамилиями читателей. В первую строку внести имя заполняемого поля – Фамилия, во вторую – фиктивного читателя Хранилище, далее внести произвольные фамилии (не менее 10). Выделить заполненный диапазон. Установить курсор в окно имен и ввести с клавиатуры имя диапазона: СписокЧит (без пробела!), нажать <Enter>. Сохранить таблицу в той же папке, что и создаваемая БД;

б) войти в Access. Щелкнуть кнопку «Внешние данные» – «Импорт» – [Excel]. В окне диалога (рис. 3.5) щелкнуть [Обзор], отыскать и выделить щелчком созданную таблицу Excel. Щелкнуть [Открыть]. В строке «Имя файла» появится описание файла таблицы Excel. Переключатель установить на позицию «Добавить копию записей в конец таблицы» и щелкнуть [ОК],

после чего по требованию СУБД сохранить и закрыть перед импортом таблицу «Список читателей»;

в) в первом из диалоговых окон настройки импорта выбрать нужный фрагмент для импорта – установить переключатель на позицию «Именованные диапазоны» (т.к. в таблице Excel есть только один такой диапазон, он будет предложен по умолчанию);

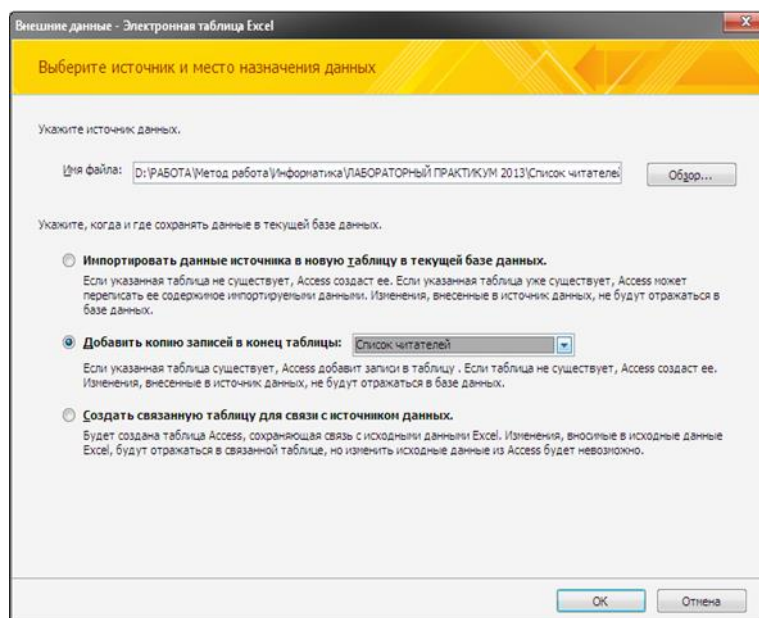


Рис. 3.5. Импорт таблицы Excel

г) второе окно следует пропустить, щелкнув [Далее] (это окно служит для возможного импорта имени поля из таблицы Excel; в нашей БД имя поля уже задано);

д) в следующем окне щелкнуть [Готово]. Предложение СУБД сохранить шаги импорта игнорировать.

17. Вновь открыть таблицу «Список читателей». Убедиться, что поля «№» и «Фамилия» заполнены. Заполнить поле «Группа», используя список подстановок. Распределить шифры групп так, чтобы среди читателей оказались представители всех групп. В поле «Группа» для читателя «Хранилище» внести прочерк (через список подстановок!). Сохранить таблицу.

Создание таблицы «Список книг»

18. Создать таблицу «Список книг». Назначить свойства полей согласно табл. 3.2. Ограничения доменов полей «Год» и «Поступила» не устанавливать. Создать необходимые списки подстановок для полей «Тема» и «Читатель».

19. Заполнить таблицу согласно индивидуальному варианту задания (табл. 3.7). Поле «Читатель» не заполнять. Даты поступления выбрать произвольно, равномерно распределив их по годам после 2005 г.

При внесении в таблицу нескольких экземпляров одной и той же книги использовать буфер обмена данными. Характеристики первой книги вносятся вручную с клавиатуры. Затем запись выделяется (щелчком по левой границе таблицы) и копируется в буфер (<Ctrl>+<C>). После этого выделяется пустая строка очередной записи (помечена звездочкой в левом столбце) и производится вставка из буфера (<Ctrl>+<V>).

20. После ввода данных о книгах заполнить поле «Читатель». Читателей указать произвольно из созданного списка подстановок (в т.ч. и читателя № 1 – «Хранилище»).

Сохранить таблицу.

Таблица 3.6

Список книг для заполнения таблиц данных

№	Авторы	Название	Место издания	Год	Стр.	Цена	Тема
1	Кузнецов В.В.	Физиология растений	М.: Высшая школа	2005	736	758 р.	Биология
2	Теллер Е.З.	Практикум по микробиологии	М.: Дрофа	2004	256	77 р.	Биология
3	Касаткина Н.В.	Физическая химия	М.: РИОР	2007	251	100 р.	Химия
4	Артеменко А.И.	Органическая химия	М.: Высшая школа	2005	605	344 р.	Химия
5	Борисенко В.В.	Основы программирования	М.: ИНТУ-ИТ	2005	328	240 р.	Программирование
6	Непейвода Н.Н.	Стили и методы программирования	М.: ИНТУ-ИТ	2005	320	300 р.	Программирование
7	Уокенбах Дж.	Диаграммы в EXCEL	М.: Вильямс	2003	448	195 р.	Офисные программы
8	Харт-Дэйвис Г.	Excel 2007. Полное руководство	М.: NT Press	2008	560	598 р.	Офисные программы

Продолжение таблицы 3.6

9	Харрингтон Д.	Проектирование реляционных баз данных	М.: Лори	2006	230	480 р.	Базы данных
10	Кронан Дж.	MS Access2007	М.: NT Press	2009	384	384 р.	Базы данных
11	Карташев А.П.	Математический анализ	СПб.: Лань	2007	448	394 р.	Математический анализ
12	Тер-Крикоров А.М.	Курс математического	М.: Физматлит	2003	672	178 р.	Математический анализ
13	Зельдович Я.Б.	Элементы прикладной математики	СПб.: Лань	2002	592	215р.	Прикладная математика
14	Пирумов У.Г.	Численные методы	М.: Дрофа	2004	224	73 р.	Прикладная математика
15	Гумилев Л.Н.	От Руси к России	М.: АСТ	2006	559	720 р.	История
16	Гумилев Л.Н.	Древняя Русь и Великая степь	М.: Айрис-пресс	2005	768	480 р.	История
17	Лем С.	Философия случая	М.: АСТ	2005	767	349 р.	Философия
18	Алексеев П.В.	История философии	М.: Проспект	2006	237	90 р.	Философия
19	Маталин А.А.	Технология машиностроения	СПб.: Лань	2010	512	430 р.	Машиностроение

Продолжение таблицы 3.6

20	Зуев Ф.Г.	Подъемно-транспортные	М.: КолосС	2006	471	380 р.	Машиностроение
21	Свитцов А.А.	Введение в мембранные технологии	М: ДеЛи Принт	2007	208	350 р.	Химическая и пищевая технология
22	Шуманн Г.	Безалкогольные напитки	СПб.: Профессия	2004	506	750 р.	Химическая и пищевая технология
23	Скотт Р.	Производство сыра	СПб.: Профессия	2005	464	748 р.	Химическая и пищевая технология
24	Малова Н.Д.	Системы вентиляции и кондиционирования	М.: ТермоКул	2005	304	715 р.	Строительство
25	Плаксин Ю.М,	Основы инженерного	М.: КолосС	2007	198	230 р.	Строительство
26	Гиргидов А.Д.	Механика жидкости и газа (гидравлика)	СПб.: СПбГТУ	2004	545	650 р.	Механика
27	Лойцянский Л.Г.	Механика жидкости и газа	М.: Дрофа	2003	840	290 р.	Механика
28	Пригожин И.	Современная термодинамика	М.: Мир	2002	461	335 р.	Теплофизика
29	Карышев А.К.	Теплофизика	М.: МГТУ	2002	108	74 р.	Теплофизика

Таблица 3.7

Варианты заполнения таблиц данных

Замечание: Номера в таблице указывают книги из списка, приведенного в табл. 3.6. Если в скобках после номера стоит число, то оно показывает количество экземпляров данной книги, которое должно быть учтено в таблице (в произвольном порядке следования записей).

Вариант	Книги
1	5, 11, 14, 15 (2), 18, 19, 21 (2), 26 (2), 29,
2	12, 7, 3 (2), 4, 1, 16, 17 (2), 23 (2), 25
3	11, 12 (2), 6 (2), 8, 9, 27 (2), 28, 20 (2)
4	11, 13, 8 (2), 26, 28, 4 (2), 2, 15, 24 (2)
5	11 (2), 13 (2), 14, 5, 6 (2), 27, 19 (2), 20
6	12, 7, 8, 9 (2), 28, 2 (2), 3, 17 (2), 25 (2)
7	11 (2), 8 (2), 1, 2, 16, 17, 22 (2), 23, 25
8	12, 13 (2), 5, 6, 9, 10 (2), 26, 27, 18, 20
9	12 (2), 14, 5 (2), 6, 8, 27, 28, 3, 16 (2)
10	1 (2), 2, 3, 4 (2), 7, 8, 11, 15, 18, 24
11	4, 10, 12, 13, 17, 22 (2), 23, 26, 28, 29 (2)
12	14 (2), 5, 26, 27, 29, 3, 15, 19 (2), 21 (2)
13	6, 7, 10, 11 (2), 14 (2), 16, 18, 20 (2), 26
14	3 (2), 5, 12, 13, 19 (2), 21, 23 (2), 27, 29
15	1, 2 (2), 8, 9, 15 (2), 18, 22 (2), 24, 29

Задания для самостоятельной работы. Часть II.

1. Создать простую форму для ввода данных в таблицу «Список книг». Сохранить ее под именем «Новая книга».

2. В режиме Конструктора отредактировать форму «Новая книга». Задать для двух элементов формы цвет шрифта и фона, а также значение свойства «Оформление» согласно индивидуальному варианту задания (табл. 3.8). Создать заголовок формы – «Новая книга» – и примечание формы. В примечании записать «Разработчик: ...» и указать свою фамилию и шифр группы.

3. Используя созданную форму, добавить в БД 3 новые книги согласно индивидуальному варианту из табл. 3.9.

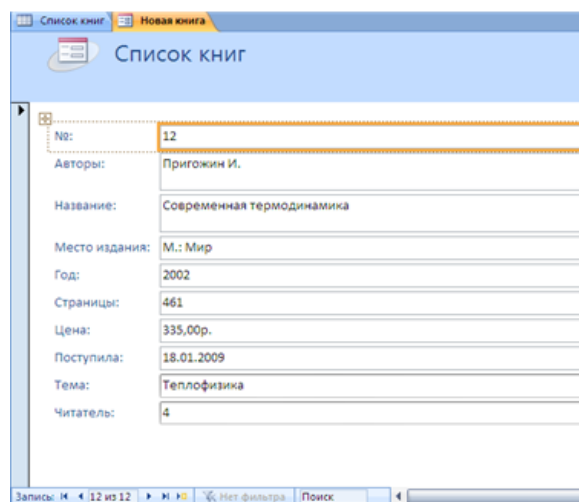
4. Сохранить обновленную таблицу. Открыть ее и убедиться в том, что новые записи в ней сохранились.

Порядок выполнения.

Создание простой формы для ввода данных

1. Открыть таблицу «Список книг» двойным щелчком по ее имени в области переходов.

2. Щелкнуть клавишу «Создание» – «Формы» – [Форма]. На правой панели появляется шаблон формы (рис. 3.6). Он отображает поля записи открытой таблицы в удобном для чтения виде. По умолчанию заголовок формы совпадает с заголовком таблицы.



The screenshot shows a form titled 'Список книг' (Book List) with the following fields and values:

Поле	Значение
№:	12
Авторы:	Пригожин И.
Название:	Современная термодинамика
Место издания:	М.: Мир
Год:	2002
Страницы:	461
Цена:	335,00р.
Поступила:	18.01.2009
Тема:	Теплофизика
Читательс:	4

Рис. 3.6. Простая форма для таблицы «Список книг»

В нижней части окна формы видна панель навигации (рис. 3.7). На ней имеются окно ввода, в котором показан номер записи, отображенной на форме, и кнопки навигации для перехода на одну запись вверх или вниз по таблице, для перехода на первую или последнюю заполненную запись и для перехода на новую (пустую) запись для ввода данных в поля. Кроме того, можно перейти на любую запись, введя с клавиатуры ее номер в окно ввода.

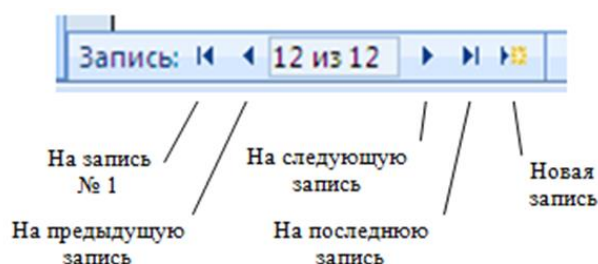


Рис. 3.7. Панель навигации формы «Список книг»

3. Сохранить форму под именем «Новая книга». Для этого в меню кнопки [Office] выбрать команду «Сохранить как» и указать в окне диалога нужное имя.

Редактирование формы в режиме Конструктора

4. Для перехода в режим Конструктора щелкнуть правой кнопкой заголовок формы – «Новая книга» – и в контекстном меню выбрать «Конструктор». Появляется окно Конструктора форм (рис. 3.8). В нем схематически показаны присутствующие на форме надписи с именами полей и окна ввода данных в эти поля. Справа видна страница свойств выделенного элемента. Она открывается щелчком по клавише «Инструменты конструктора форм» – «Конструктор» – «Сервис» – [Страница свойств] или после двойного щелчка по выбранному элементу.

Вставляя курсор в окно того или иного свойства выбранного элемента, можно изменить значение этого свойства. Исключение составляют горизонтальные размеры элементов. Если форма создается автоматически на базе конкретной таблицы, размеры меняются только для всех надписей или для всех окон ввода одновременно. Взаимное расположение элементов также менять нельзя, они как бы «привязаны» друг к другу. Если форма создается в режиме Мастера или Конструктора, это ограничение снимается.

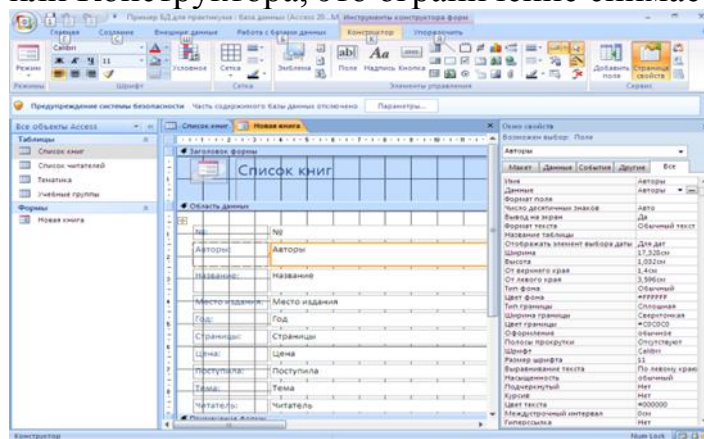


Рис. 3.8. Конструктор форм

Для установки значений свойств элементов во многих окнах имеются готовые списки значений. При выборе цветов шрифта, фона и т.п. таких списков два. Обычный знак списка (черный треугольник) открывает сокращенный набор шаблонных вариантов. Значок, помеченный многоточием, открывает расширенную палитру для выбора цвета.

5. Выделяя по очереди отдельные элементы формы, изменить для них значения свойств согласно индивидуальному варианту задания (табл. 3.8). Сохранить изменения.

6. Установить курсор в окно заголовка формы и заменить его на «Новая книга». Установить для заголовка выравнивание по центру, шрифт Times New Roman, кегль 24, синие буквы на желтом фоне.

7. В нижней части окна Конструктора найти заголовок «Примечание формы». Мышью отодвинуть границу области примечания вниз, создав полосу для размещения примечания. В разделе «Инструменты конструктора форм» – «Конструктор» – «Элементы управления» найти и щелкнуть элемент «Надпись». Указателем мыши (он принимает вид черного крестика) обвести

на полосе контур будущего примечания. В получившееся окно вставить курсор и набрать текст примечания. Оформить примечание по своему усмотрению. Сохранить форму.

8. Закрыть таблицу «Список книг». Щелкнуть по названию формы «Новая книга» в области переходов и во всплывшем меню выбрать команду «Открыть» (форма открывается в режиме формы, см. рис. 3.6). С помощью формы внести в БД три новые записи согласно индивидуальному варианту задания (табл. 3.9).

Данные, введенные в какое-либо окно ввода в форме, заносятся в таблицу сразу же, как только курсор переводится в следующее окно, поэтому после завершения ввода записей надо просто закрыть форму.

9. Открыть таблицу «Список книг» и убедиться, что в ней появились введенные записи. Сохранить таблицу.

Таблица 3.8

№	Изменяемый элемент	Цвет фона	Гарнитура шрифта	Цвет шрифта	Кегль шрифта	Начертание и насыщенность букв	Выравнивание
1	Надпись «Авторы»	Синий	Times New Roman	Белый	12	Тонкое курсивное	По центру
	Окно ввода «Тема»	Бирюзовый	Calibri	Красный	14	Полужирное	По правому краю
2	Надпись «Год»	Зеленый	Arial	Желтый	16	Плотное курсивное	Подбор
	Окно ввода «Цена»	Фиолетовый	Cambria	Светло-голубой	18	Тонкое подчеркнутое	По центру
3	Надпись «Читатель»	Красный	Courier New	Бирюзовый	12	Сверхжирное	По правому краю
	Окно ввода «Место издания»	Коричневый	Garamond	Желтый	14	Среднее курсивное	Подбор
4	Надпись «Страницы»	Желтый	Calibri	Синий	16	Тонкое	По центру
	Окно ввода «Поступила»	Голубой	Times New Roman	Зеленый	18	Полужирное курсивное	По правому краю

Продолжение Таблица 3.8

№	Изменяемый элемент	Цвет фона	Гарнитура шрифта	Цвет шрифта	Кегль шрифта	Начертание и насыщенность букв	Выравнивание
5	Надпись «Название»	Черный	Garamond	Белый	12	Полужирное подчеркнутое	Подбор
	Окно ввода «№»	Синий	Courier New	Желтый	14	Среднее подчеркнутое	По центру
6	Надпись «Читатель»	Бирюзовый	Cambria	Черный	16	Тонкое курсивное	По правому краю
	Окно ввода «Страницы»	Зеленый	Arial	Красный	12	Полужирное	Подбор
7	Надпись «Тема»	Желтый	Times New Roman	Коричневый	14	Плотное курсивное	По центру
	Окно ввода «Цена»	Красный	Calibri	Белый	18	Сверхжирное подчеркнутое	По правому краю
8	Надпись «Поступила»	Синий	Arial	Белый	16	Сверхжирное	Подбор
	Окно ввода «Поступила»	Коричневый	Cambria	Голубой	14	Среднее курсивное	По центру

Продолжение Таблица 3.8

№	Изменяемый элемент	Цвет фона	Гарнитура шрифта	Цвет шрифта	Кегль шрифта	Начертание и насыщенность букв	Выравнивание
9	Надпись «Цена»	Черный	Courier New	Желтый	12	Тонкое	По правому краю
	Окно ввода «Тема»	Синий	Garamond	Коричневый	16	Полужирное курсивное	Подбор
10	Надпись «Страницы»	Бирюзовый	Calibri	Красный	12	Полужирное подчеркнутое	По центру
	Окно ввода «Читатель»	Зеленый	Times New Roman	Белый	14	Среднее подчеркнутое	По правому краю
11	Надпись «Год»	Красный	Garamond	Красный	18	Тонкое курсивное	Подбор
	Окно ввода «№»	Коричневый	Courier New	Коричневый	16	Полужирное	По центру
12	Надпись «Место»	Бирюзовый	Cambria	Бирюзовый	12	Плотное курсивное	По правому краю
	Окно ввода «Авторы»	Зеленый	Arial	Зеленый	14	Тонкое подчеркнутое	Подбор

Продолжение Таблица 3.8

№	Изменяемый элемент	Цвет фона	Гарнитура шрифта	Цвет шрифта	Кегль шрифта	Начертание и насыщенность букв	Выравнивание
13	Надпись «Название»	Желтый	Arial	Синий	18	Сверхжирное	По центру
	Окно ввода «Название»	Красный	Calibri	Бирюзовый	16	Среднее курсивное	По правому краю
14	Надпись «Авторы»	Синий	Garamond	Белый	12	Тонкое	Подбор
	Окно ввода «Место издания»	Коричневый	Cambria	Голубой	14	Полужирное курсивное	По центру
15	Надпись «№»	Серый	Courier New	Красный	14	Полужирное подчеркнутое	По правому краю
	Окно ввода «Год»	Бирюзовый	Arial	Черный	12	Среднее подчеркнутое	Подбор

Дополнительные книги для внесения в таблицу

Вариант	Книги (№ по табл. 3.6)
1	7, 8, 9
2	11, 21, 27
3	10, 14, 19
4	1, 22, 25
5	9, 16, 18
6	21, 23, 29
7	7, 12, 15
8	14, 19, 24
9	4, 15, 29
10	2, 4, 8
11	11, 12, 14
12	3, 4, 23
13	12, 15, 19
14	4, 20, 22
15	1, 16, 21

Задания для самостоятельной работы. Часть III.*Теоретические положения**Запросы*

Запросы представляют собой основной тип документов БД. Большинство форм и отчеты создаются на основании запросов.

Различные формы запросов позволяют проводить основные действия по обработке данных:

а) *запрос на выборку* служит для поиска в БД записей, удовлетворяющих указанному в запросе условию;

б) *запрос на обновление* позволяет автоматически изменить какие-то данные в записях БД по заданному правилу;

в) *запрос на удаление* служит для удаления из БД записей, удовлетворяющих указанному условию;

г) *запрос на создание* таблицы позволяет создать новую таблицу БД на основе имеющихся таблиц и т.д.

Различают статические и параметрические запросы. В статических запросах условия отбора записей для обработки задаются в самом запросе. В параметрических запросах условия отбора вводятся пользователем с клавиатуры по ходу выполнения запроса.

Создавать запросы мы будем сразу в режиме Конструктора. Возможности Мастера запросов ограничены и здесь не рассматриваются.

Выражения. Построитель выражений

Выражения применяются для составления ограничений домена атрибута, критериев отбора, обновления данных и создания вычисляемых полей в запросах. Выражения условий и критериев отбора имеют логический тип. Выражения для вычисления могут иметь любой тип, но он обязан соответствовать типу поля, для которого строится выражение.

Если выражение служит для вычисления и находится в ячейке, соответствующей какому-то вычисляемому полю (при обновлении, например), то по умолчанию подразумевается наличие операции присваивания := (см., например, рис. 3.13)

{поле :=} выражение

Здесь и далее фигурными скобками выделяются элементы условий и выражений, которые не записываются, но их наличие в данном месте подразумевается.

Если выражение является ограничением домена или критерием, который надо применить к какому-то полю, то оно включает операторы отношений (символы равенств и неравенств). По умолчанию считается, что слева от такого оператора стоит имя поля (рис.3.13).

Выражение состоит из операндов и операторов. *Операнды* – это слагаемые, множители, основания и показатели степени, аргументы функций. Операторы соединяют операнды, образуя выражения. Кроме того, в выражениях можно использовать круглые скобки для образования сложных операндов.

Элементами операндов могут быть:

а) константы любого типа – числа, даты, логические значения **Истина** и **Ложь**, тексты, набранные с клавиатуры. Текстовые константы заключаются в кавычки, даты обрамляются символами #.

б) имена полей БД. Они должны быть обрамлены квадратными скобками. Общая структура имени поля произвольной таблицы:

[имя таблицы] ! [имя поля]

в) функции.

Операторы делятся на группы:

- арифметические: +, −, *, /, \ (деление целых чисел нацело с отбрасыванием остатка), Mod (определение остатка от деления нацело), ^ (возведение в степень);
- операторы сравнения (=, <, >, >=, <=, >= и др.). Они применяются при создании ограничений и критериев;
- логические (And, Or, Not и др.);
- оператор Like, используемый при создании условий для текстовых полей в комбинации с символами создания текстовых масок «?» (заменяет собой любой одиночный символ), и «*» (заменяет любую группу символов). Например, условие

Like " A* " (3.1)

служит для отбора всех значений текстового поля, начинающихся с буквы A;

- оператор Between ... And ... служит для отбора значений полей, лежащих между указанными в операторе границами. Например, оператор

{Поле} Between 300 And 500 (3.2)

эквивалентен условию

{поле} >= 300 And {поле} <= 500 (3.3)

- оператор & создает сцепление двух текстовых строк; например, выражение "студент"&"ов" эквивалентно слову "студентов".

Существуют и другие операторы.

Для облегчения работы в СУБД Access имеется специальный инструмент – Построитель выражений. Он запускается кнопкой на возникающей в соответствующем случае контекстной вкладке инструментальной ленты.

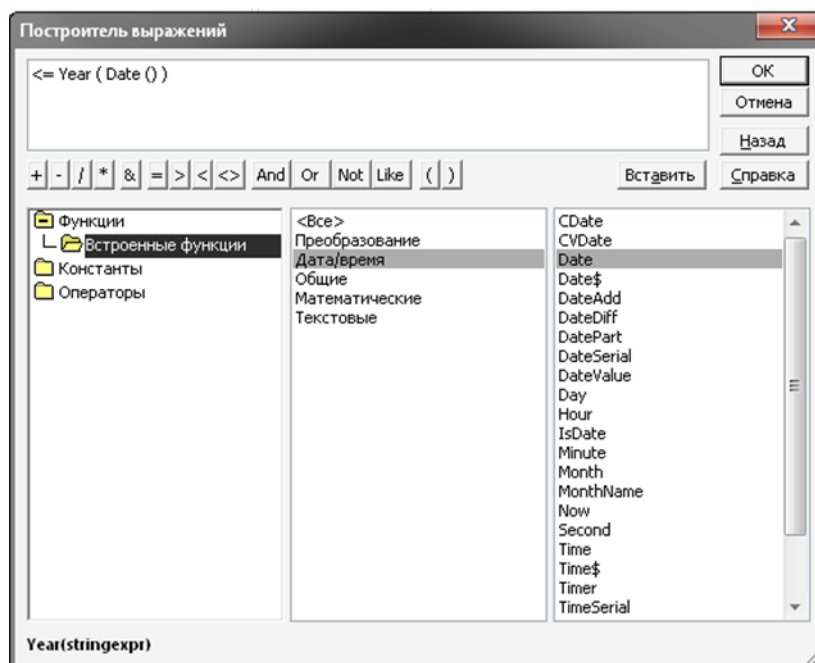


Рис. 3.9. Построитель выражений в условиях (заменить)

Например, в нашей таблице «Список книг» должно существовать ограничение домена для значений поля «Год» – заносимое в таблицу значение года издания книги не может оказаться больше текущего календарного года. Для создания такого ограничения надо открыть таблицу «Список книг» в режиме Конструктора и установить курсор в ячейку «Условие на значение» для поля «Год». На контекстной вкладке «Конструктор» (раздел «Сервис») имеется кнопка [Построитель]. После щелчка по ней открывается окно Построителя выражений (рис. 3.9).

В верхней части находится окно ввода, в котором строится выражение. Под ним расположены три окна со списками элементов для построения. В первом окне выбирается вид элемента – встроенная функция, константа или оператор; во втором – категория (разновидность) элементов выбранного вида; в третьем – конкретный элемент.

Для вставки выбранного элемента в выражение надо выделить его и щелкнуть [Вставить] либо дважды щелкнуть мышью строку с его названием или обозначением. Кроме этого, в окне построения выражений можно вводить отдельные элементы выражения с клавиатуры.

На рис. 3.9 показано построение ограничения на значение поля «Год». Оно строится в три шага:

- 1) в списке операторов выбирается неравенство <=;
- 2) в списке встроенных функций категории «Дата/Время» выбирается функция Year(), определяющая целое число – год, соответствующий дате, указанной в качестве аргумента;
- 3) в окне выражения курсор устанавливается внутрь скобок аргумента функции Year() (если в скобках по умолчанию помещен какой-то текст, его надо удалить) и в списке функций категории «Дата/Время» выбирается

функция Date(), автоматически вычисляющая текущую календарную дату по показаниям системного календаря компьютера. Эта функция не имеет аргумента.

Полученное условие имеет вид

$$\leq \text{Year}(\text{Date()}) \quad (3.4)$$

После построения выражения для условия надо щелкнуть [ОК] в окне Построителя, и выражение вставляется в строку «Условие на значение» для рассматриваемого поля таблицы. В соседнюю строку «Сообщение об ошибке» надо ввести в кавычках текст сообщения (например: "Неверное значение!") для пользователя БД на случай, если он при заполнении БД введет ошибочное значение года.

Задания для самостоятельной работы часть III

1. Составить условия на значения и сообщения об ошибках для полей «Год» (по описанному примеру) и «Поступила» (самостоятельно) таблицы «Список книг», соответствующие табл. 3.2.

Открыть форму «Новая книга» и внести в таблицу любую книгу из табл. 3.6, указав год издания и дату поступления, соответствующие будущему времени. Убедиться, что СУБД не позволяет внести в таблицу неверные значения.

2. Составить статический запрос на выборку согласно индивидуальному варианту (табл. 3.10). Сохранить его под именем «Статический запрос работы 3».

3. Составить и выполнить запрос на создание таблицы, содержащей исходные цены книг (по описанному примеру). Сохранить запрос. Открыть и просмотреть созданную таблицу.

4. Составить и выполнить запрос на обновление значений цены в таблице «Список книг» согласно индивидуальному варианту (табл. 3.11). Сохранить и выполнить запрос.

Убедиться, что таблица «Исходные цены книг» сохранила первоначальные значения цен, а в таблице «Список книг» цены изменились.

5. Используя форму «Новая книга», внести в таблицу «Список книг» любую книгу из табл. 3.6, указав в качестве даты поступления текущее число. Убедиться, что таблица «Исходные цены книг» не содержит записи о новой книге.

6. Составить и выполнить запрос на добавление записей в таблицу «Исходные цены книг» (по описанному примеру). Убедиться, что в таблице «Исходные цены книг» появилась запись о новой книге.

Порядок выполнения работы (п. 2–6 задания). Создание статического запроса на выборку

1. Щелкнуть кнопку «Создание» – «Другие» – [Конструктор запросов]. На правой панели открывается вкладка «Запрос 1», поверх которой появляется окно диалога «Добавление таблицы». В нем из списка имеющихся таблиц БД надо выбрать таблицы, которые потребуются для создания запроса.

В качестве примера рассмотрим запрос, при выполнении которого будет получен список имеющихся в библиотеке книг по химии или биологии. Для его создания требуется одна таблица – «Список книг», которую и следует выбрать, затем щелкнуть [Добавить] и закрыть окно. На ленте появляется контекстная вкладка «Конструктор запросов» (рис. 3.10). Тип запроса выбирается в одноименном разделе вкладки. По умолчанию предлагается создание запроса на выборку (тип запроса – «Выбор»).

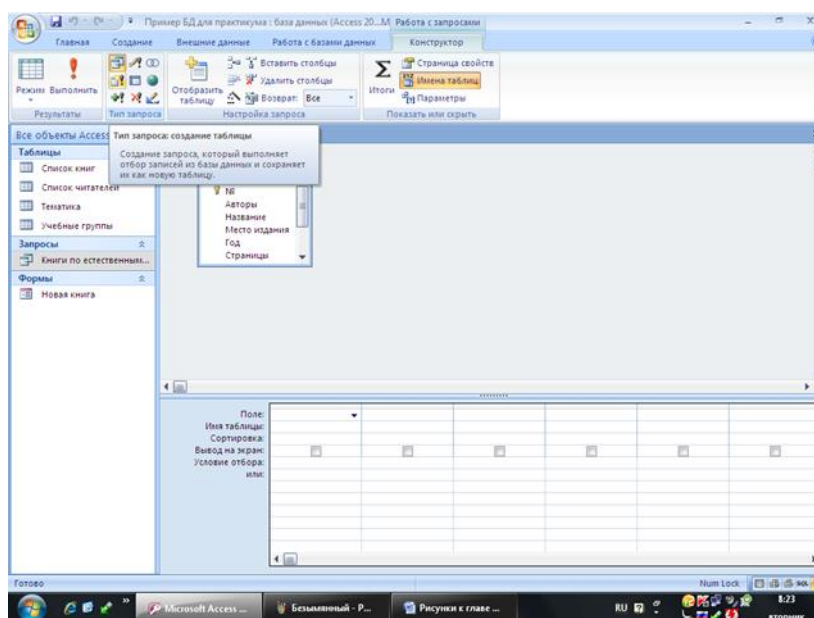


Рис. 3.10. Конструктор запросов

В режиме Конструктора запросов на правой панели видны две области. В верхней – «области таблиц» – отображается список таблиц, выбранных для создания запроса, их структура и связи между ними. В нашем случае здесь видна одна таблица «Список книг». В нижней – «области критериев» – видна пустая сетка с подписями строк.

2. В первой строке области критериев указать заголовки полей, которые требуется отобразить в запросе. Для этого мышью перетащить названия полей со схемы таблицы в нужные окна либо выбрать нужное поле из общего списка полей всех выбранных таблиц (значок списка появляется в ячейке после помещения в нее курсора). Во второй строке автоматически отображается имя таблицы, из которой взято поле.

3. В третьей строке указать способ сортировки по значениям того или иного поля.

4. В четвертой строке флажками пометить поля, которые будут выведены на экран при выполнении запроса.

5. В последующих строках сформировать условия отбора. Условие для числового поля имеет вид равенства или неравенства и выглядит аналогично (3.4). Для создания условия-равенства, налагаемого на текстовое поле, достаточно ввести с клавиатуры в ячейку сетки нужный текст в кавычках (если забыть про кавычки, Access добавит их самостоятельно после перехода к другой ячейке), или создать выражение текстового типа с помощью Построителя.

При этом отдельные условия, размещенные в одной строке запроса в разных столбцах, считаются объединенными операцией логического умножения (конъюнкции) AND, а условия в разных строках – операцией логического сложения (дизъюнкции) OR. Если надо объединить операцией AND два условия для одного и того же поля, сложное условие (логическое выражение) записывается в одной ячейке сетки. Например, чтобы выбрать из БД записи о книгах, содержащих от 300 до 500 страниц, можно для поля «Страницы» задать условие отбора либо в виде (3.2), либо в виде, соответствующем (3.3):

$$\geq 300 \text{ AND } \leq 500. \quad (3.5)$$

Для запроса, рассматриваемого в качестве примера, условием будут слова «Химия» и «Биология», записанные в сосед-них строках поля «Тема» (рис. 3.11).

6. Сохранить запрос, щелкнув по кнопке сохранения на панели быстрого доступа и указав в окне диалога имя запроса. Для выполнения запроса щелкнуть на вкладке «Конструктор» кнопку [Выполнить]. На экране возникает таблица, содержащая результаты поиска и выбора нужных книг.

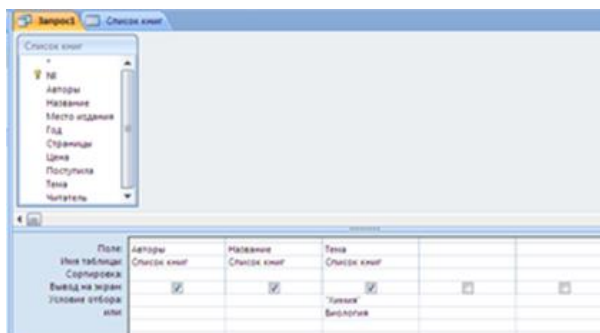


Рис. 3.11. Структура запроса на выборку

Эта таблица является временной и в БД не сохраняется. Сохраняется только запрос, который при необходимости каждый раз выполняется заново.

Для сохранения результатов выполнения запроса надо либо преобразовать его в запрос на создание новой таблицы (см. далее), либо оформить на основе полученных результатов отчет.

Формирование запроса на создание таблицы

При работе с БД в таблицу могут вноситься различные изменения. При этом может оказаться необходимым сохранить («заморозить») какое-либо промежуточное состояние таблицы или ее части. Например, может возникнуть необходимость сохранить исходные значения цен книг на момент поступления в библиотеку. Для этого можно автоматически создать «урезанную» таблицу, содержащую минимально необходимый набор полей, например «№» (ключевое поле счетчика), «Авторы», «Название» и «Цена» (для указанной цели достаточно иметь всего два поля – «№» и «Цена», но такая таблица потеряла бы наглядность).

Для создания таблицы используется специальный запрос, формируемый в следующем порядке:

7. Создать запрос на выборку, который послужит основой для преобразования в запрос на создание таблицы. В рассматриваемом примере в запрос включаются четыре перечисленных поля, а условия отбора отсутствуют.

8. Щелкнуть кнопку «Конструктор» – «Тип запроса» – [Тип запроса: создание таблицы]. Окно Конструктора принимает вид, показанный на рис. 3.12.

9. В окне диалога указать имя создаваемой таблицы – «Исходные цены книг». Переключатель установить в положение «в текущей базе данных» (как видно из структуры окна, новую таблицу на основе данной БД можно создать и в другой базе).

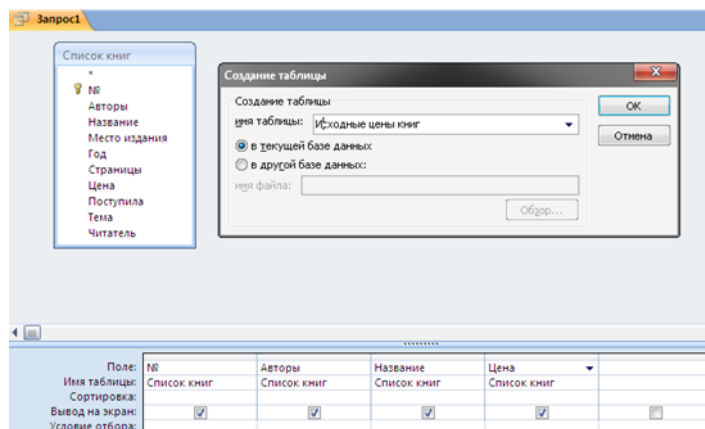


Рис. 3.12. Запрос на создание таблицы

10. Щелкнуть [OK].

11. Сохранить запрос под именем «Создание таблицы». 12. Запустить запрос на выполнение, щелкнув кнопку

«Конструктор» – [Выполнить].

Сохраненный запрос можно использовать впоследствии для обновления созданной таблицы (например, в случае изменения каких-то данных в исходной таблице). Для этого после внесения изменений в исходную таблицу («Список книг») надо открыть запрос и запустить его на выполнение. В результате на месте старой таблицы с помощью запроса будет создана новая таблица той же структуры, в которой будут отражены изменения, сделанные в исходной таблице.

Создание запроса на обновление данных

Запрос на обновление позволяет избежать трудоемкой работы по ручной корректировке данных, если эта корректировка должна быть произведена по некоторому единому правилу.

Рассмотрим для примера ситуацию – вследствие износа и морального старения все книги, выпущенные до 2005 года, были уценены на 10 %. Необходимо внести в таблицу соответствующие изменения.

Идея запроса на обновление заключается в том, что выбирается поле, данные которого должны быть обновлены, и задается новое значение поля или выражение для его вычисления. При необходимости задается условие отбора записей, данные которых должны быть изменены. Запрос создается в следующем порядке:

13. Создать запрос на выборку на основе таблицы, в которую будут вноситься изменения («Список книг»). Для запроса использовать два поля таблицы – «Год» (для создания условия отбора записей) и «Цена» (для изменения).

14. В строку «Условие отбора» для поля «Год» внести условие отбора книг, изданных до 2005 года: <2005. Это можно сделать прямо в ячейке сетки с помощью клавиатуры.

15. Щелкнуть кнопку «Конструктор» – «Тип запроса» – [Запрос на обновление]. Окно Конструктора принимает вид, показанный на рис. 3.13.

16. В строку «Обновление» для поля «Цена» с помощью Построителя («Работа с запросами» – «Конструктор» – «Настройка запроса» – [Построитель]) ввести новое выражение цены – в рассматриваемом примере она должна составлять 90 % от прежней цены.

17. Сохранить запрос под именем «Обновление цены». 18. Запустить запрос на выполнение.

При выполнении запроса СУБД выводит на экран окно с предупреждением о том, что в таблице будет изменено соответствующее количество записей (рис. 3.13). Если в окне щелкнуть [Да], то изменения будут внесены, причем это действие уже невозможно будет отменить.

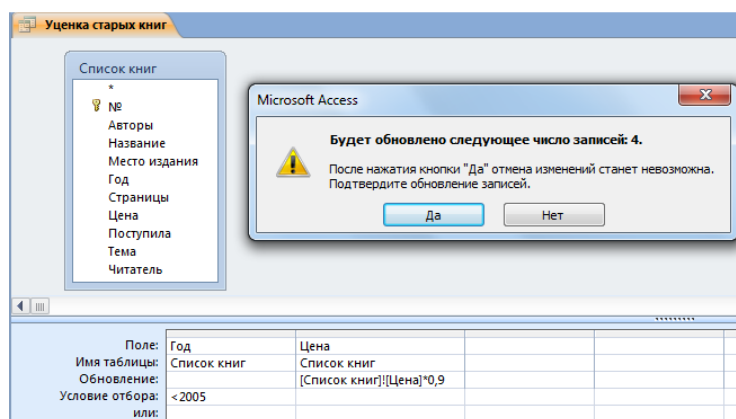


Рис. 3.13. Создание запроса на обновление

Создание запроса на добавление записей

Использование запроса на создание таблицы для обновления записей в таблице, ранее созданной с помощью этого запроса, может оказаться нецелесообразным. Допустим, в исходной таблице «Список книг» были автоматически обновлены значения поля «Цена», а уже после этого в библиотеку поступили новые книги, данные о которых были занесены в эту таблицу. Если теперь обновить таблицу «Исходные цены книг» с помощью существующего запроса на создание таблицы, то для старых книг в нее попадут уже не исходные, а измененные значения цены.

Значит, надо иметь возможность добавлять в таблицу «Исходные цены книг» только данные о новых книгах, оставляя старые без изменения. Для этого служит запрос на добавление записей (рис. 3.14), создаваемый в следующем порядке:

19. Создать запрос на выборку на основе таблицы «Список книг», включающий поля, которые соответствуют таблице «Исходные цены книг», а также поле «Поступила», необходимое для создания условия отбора. Условием является совпадение даты поступления книги в библиотеку с текущей календарной датой (см. рис. 3.14), поскольку добавление записей в таблицу исходных цен должно выполняться сразу же после занесения книги в основную таблицу «Список книг» – в тот же день.

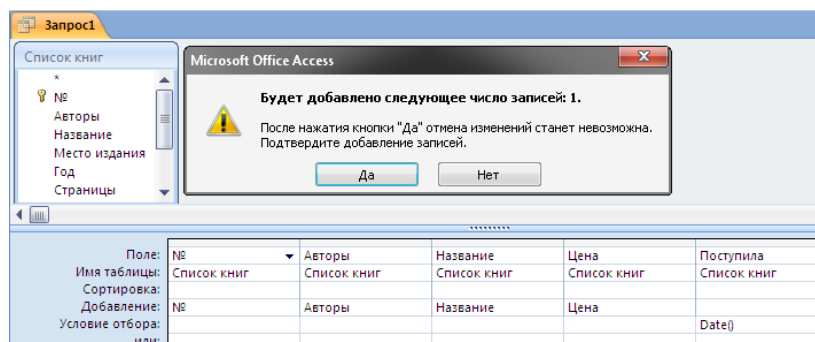


Рис. 3.14. Запрос на добавление записей

20. Щелкнуть кнопку «Конструктор» – «Тип запроса» – [Тип запроса: добавление]. Окно Конструктора принимает вид, аналогичный показанному на рис. 3.12, но в окне диалога требуется теперь указать имя таблицы, в которую будет добавлена запись.

21. Щелкнуть [ОК]. В области критериев появляется строка «Добавление» (рис. 3.14). Если среди полей в этой строке указано поле, которое отсутствует в таблице, куда добавляются записи (например, поле «Поступила»), надо установить в ячейку курсор и удалить имя этого поля.

22. Сохранить запрос.

23. Запустить запрос на выполнение; в окне предупреждения (рис. 3.14) щелкнуть [Да].

Таблица 3.10

Варианты запросов на выборку

№	Поля, включенные в запрос	Условия отбора	Поле и тип сортировки	Примечание
1	Авторы, Название, Год издания, Тема	Тема: «Программирование» или «Офисные программы»; изданы после 2004 г.	Год издания, убывание	Поле «Тема» на экран не выводить
2	Авторы, Название, Страницы, Поступила	Больше 300 стр., поступили до 1.01.2010	Страницы, убывание	
3	Авторы, Название, Место издания, Поступила	Книги по математическому анализу, изданные до 2005 г.	Поступила, возрастание	
4	Авторы, Название, Год издания, Тема	Более 500 стр., изданы после 2004 г.	Год издания, убывание	Поле «Тема» на экран не выводить
5	Авторы, Название, Страницы, Год издания	Книги издательства «СПб.: Лань», изданы после 2006 г.	Год издания, возрастание	
6	Авторы, Название, Цена, Тема	Книги по темам «Офисные про-граммы» или «Базы данных», дешевле 500 р.	Авторы, убывание	Поле «Тема» на экран не выводить
7	Авторы, Название, Страницы, Год	Более 300 стр., изданы до 2006 г.	Авторы, возрастание	

Окончание табл. 3.10

8	Авторы, Название, Страницы, Тема	Книги по темам «Прикладная математика» или «Математический анализ», более 500 стр.	Страницы, убывание	Поле «Тема» на экран не выводить
9	Авторы, Название, Страницы, Тема	Место издания: «М.: ИНТУИТ», больше 320 стр.	Дата поступления, возрастание	Поле «Издательство» на экран не выводить
10	Авторы, Название, Год издания, Страницы	Изданы после 2005 г., более 500 стр.	Название, возрастание	
11	Авторы, Название, Год издания, Поступила	Изданы более, чем за два года до поступления в библиотеку, или поступили до начала 2008 г.	Авторы, возрастание	
12	Авторы, Название, Цена, Тема	Авторы с фамилиями на буквы «Г» или «С», дороже 500 р.	Цена, возрастание	Поле «Тема» на экран не выводить
13	Авторы, Название, Страницы, Цена	Больше 300 стр. и дороже 300 р.	Авторы, возрастание	
14	Авторы, Название, Место издания, Тема	Темы «Машиностроение», «Химическая и пищевая технология»; изданы после 2005 г.	Название, убывание	Поле «Тема» на экран не выводить
15	Авторы, Название, Страницы, Цена	Авторы с фамилиями на буквы «Г» или «Х», книги толще 250 стр.	Страницы, возрастание	

Варианты обновления цены

№	Способ обновления цены
1	Увеличить на 25 р
2	Уменьшить в 1,2 раза
3	Увеличить на 15 %
4	Уменьшить на 5 %
5	Уменьшить на 20 р.
6	Увеличить в 1,05 раза
7	Увеличить на 25 р.
8	Уменьшить в 1,2 раза
9	Увеличить на 15 %
10	Уменьшить на 5 %
11	Уменьшить на 20 р.
12	Увеличить в 1,05 раза
13	Увеличить на 25 р.
14	Уменьшить в 1,2 раза
15	Увеличить на 10 %

Задания для самостоятельной работы. Часть IV.*Теоретические положения**Связывание таблиц. Схема данных*

Отдельные таблицы БД должны быть связаны между собой для того, чтобы данные из этих таблиц можно было использовать совместно.

При создании БД в первой части практической работы №3 мы использовали отдельные наборы данных из таблиц для создания списков подстановок в поля других таблиц. Тем самым были автоматически заданы связи между этими таблицами.

Структура связей отображается в Схеме данных, которая открывается щелчком по «Работа с базами данных» – «Показать или скрыть» – [Схема данных]. На схеме показаны списки полей имеющихся в БД таблиц и связи между ними (рис. 3.15).

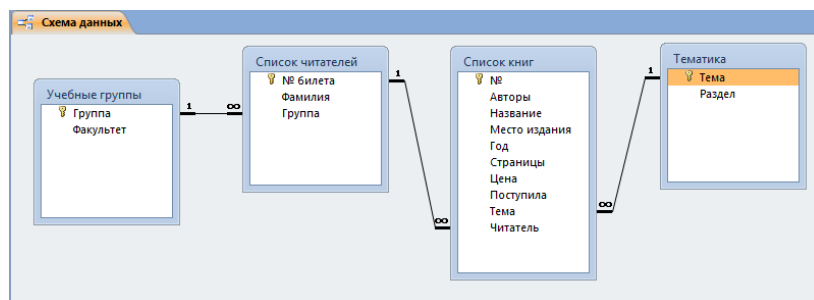


Рис. 3.15. Схема данных

Если какая-то таблица не показана на схеме, следует щелкнуть по схеме правой кнопкой мыши и в контекстном меню выбрать команду «Добавить таблицу». Добавление проводится так же, как при создании запросов. Списки полей можно перетаскивать мышью, придавая схеме данных более удобный вид.

Если какая-либо связь отсутствует (скорее всего, это значит, что при заполнении таблицы какое-то значение было занесено с клавиатуры, а не с помощью списка подстановок), ее можно установить принудительно. Для этого достаточно провести мышью с нажатой левой кнопкой от строки с именем ключевого поля в одной таблице («главной») к строке с именем поля в другой таблице («подчиненной»), которое является внешним ключом для связи. На рис. 3.15 внешние ключи помечены символом ∞.

Если символы 1 и ∞ на линиях связи, показывающие характер связи – «один ко многим», отсутствуют, следует изменить связь, щелкнув по ее линии правой кнопкой и выбрав в контекстном меню команду «Изменить связь». В окне диалога (рис. 3.16) надо установить флажок «Обеспечение целостности данных» и щелкнуть [ОК].

Целостность означает, что данные разных таблиц должны соответствовать друг другу с точки зрения структуры таблиц, связей между ними и смысла содержания полей. Если эти требования соблюдены, на линии связи появляются указанные значки. Появление сообщения об отказе в поддержании целостности означает, что данные ключевых полей связи могут не соответствовать друг другу:

а) в поле внешнего ключа подчиненной таблицы имеются значения, отсутствующие в главной таблице. Например, в поле «Читатель» таблицы «Список книг» может быть указан номер читательского билета, которого нет в таблице «Список читателей»;

б) ключевые поля связи не совпадают по типу данных. Допускается единственная возможность несовпадения типов, не нарушающая целостности – можно связывать ключевое поле типа «Счетчик» с полем числового типа «Длинное целое».

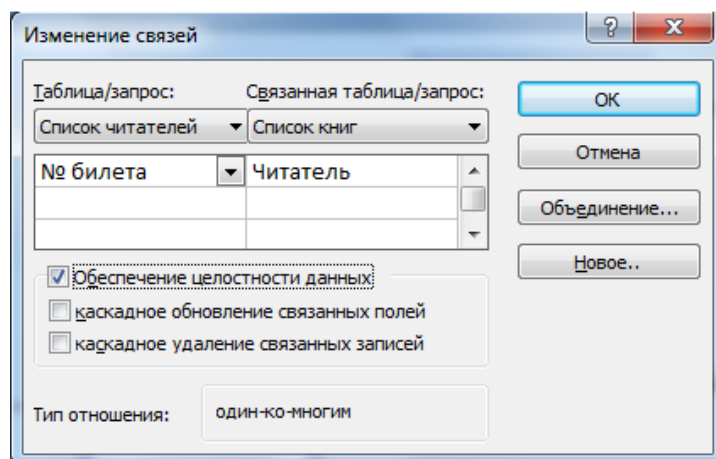


Рис. 3.16. Определение параметров связи

После установления связи между таблицами в главной таблице появляется дополнительный столбец со значками «плюс». Значки указывают на наличие подтаблиц, в которых собраны записи связанной таблицы, отвечающие данной записи главной таблицы. Подтаблица раскрывается щелчком по знаку «плюс» и закрывается щелчком по знаку «минус», появляющемуся при открытии подтаблицы.

Прежде чем создавать какие-то запросы на основе связанных таблиц, следует проверить и отредактировать схему данных.

Составление запроса на основе связанных таблиц

Последовательность действий при составлении запроса на основе нескольких связанных таблиц практически не отличается от создания запроса на основе одной таблицы. Разница состоит в том, что в область таблиц окна Конструктора надо добавить все таблицы, поля которых используются в запросе для вывода на экран или создания критериев отбора. Также надо добавить те таблицы, через которые проходят связи между таблицами, поля которых используются в запросе, даже если эти промежуточные таблицы непосредственно в запросе не используются.

Допустим, надо определить разделы тематики книг, выданных студентам механического факультета. В результате выполнения запроса на экран можно вывести всего одно поле «Раздел» из таблицы «Тематика», а для создания критерия отбора использовать поле «Факультет» из таблицы «Учебные группы». Однако из схемы данных (рис. 3.15) видно, что указанные таблицы не связаны непосредственно между собой. Цепочка связи проходит через все таблицы, поэтому их все надо добавить в запрос. Структура запроса будет выглядеть так, как показано на рис. 3.17.

Создание параметрических запросов

Статический запрос легко создать, однако его использование не является легким и удобным. Если необходимо изменить условие отбора (например, выявить тематические разделы книг, выданных студентам другого

факультета), приходится каждый раз входить в режим Конструктора и вручную изменять условие, или просто создавать новый запрос.

Избежать этих трудностей можно путем применения параметрического запроса. В таком запросе условие отбора является неопределенным параметром, т.е. задается в ходе выполнения.

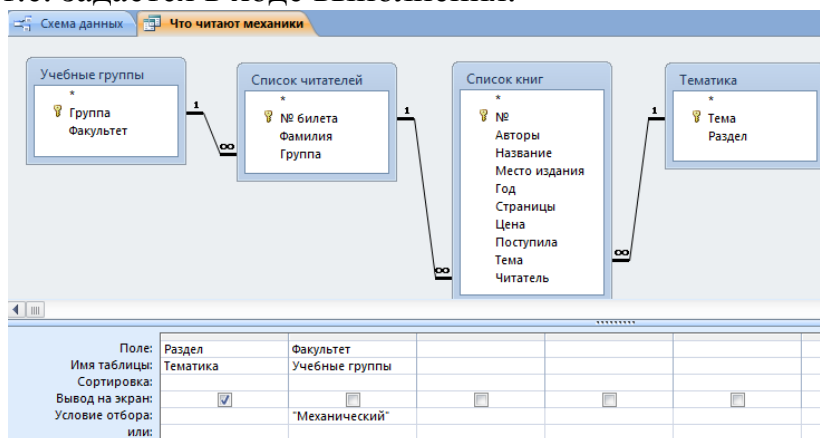


Рис. 3.17. Запрос на основе связанных таблиц

Параметром запроса может быть только константа, но не выражение. Для задания параметрического критерия отбора надо при создании запроса в режиме Конструктора вместо значения критерия ввести в квадратных скобках текст вопроса или предписания пользователю. При запуске запроса на экране появится диалоговое окно с этим текстом и окном ввода для значения параметра. После ввода значения параметра запрос будет выполнен с использованием этого значения в качестве критерия.

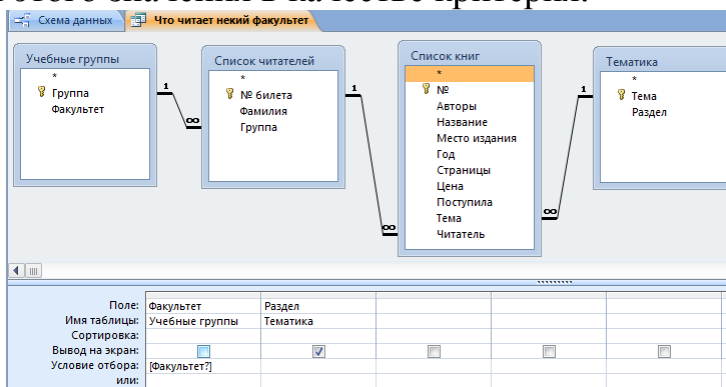


Рис. 3.18. Структура параметрического запроса

На рис. 3.18–3.19 показаны структура запроса, аналогичного описанному выше статическому, но с использованием параметрического критерия – названия факультета, и диалоговое окно ввода параметра.

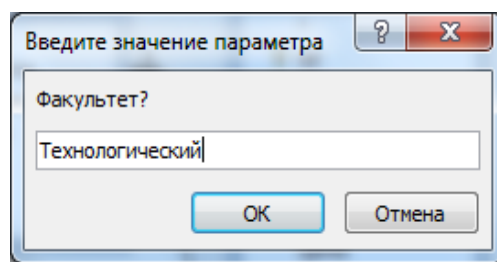


Рис. 3.19. Ввод значения параметра

Параметрический запрос может включать несколько параметров. При выполнении запроса значения этих параметров запрашиваются поочередно.

Задания для самостоятельной работы

1. Открыть схему данных. Проверить наличие связей «Один ко многим» между таблицами. Схема должна принять вид, соответствующий рис. 3.15. При необходимости установить нужные связи самостоятельно.

2. Создать на основе связанных таблиц два статических запроса на выборку согласно индивидуальному заданию (табл. 3.12, 3.13). Сохранить запросы.

Общее требование к запросам по пунктам 2, 3 настоящего задания: если в запросе требуется получить список студентов, групп или факультетов, то книги, находящиеся в хранилище, не должны отображаться в этом списке.

3. Создать параметрический запрос согласно индивидуальному заданию (табл. 3.14). Значение поля, обозначенное в задании словами «некоторый», «какой-то» и т.п., должно быть параметром запроса. Сохранить запрос.

4. Открыть схему данных. Установить связь «Один к одному» между таблицами «Список книг» и «Исходная цена книги» по ключевым полям таблиц.

5. Создать параметрический запрос на выборку, в котором будут отражены исходные цены книг, выданных какому-либо читателю (по введенному в качестве параметра значению поля «Фамилия»).

Таблица 3.12

Варианты заданий для составления статического запроса 1

Вариант	Задание: составить запрос для получения списка
1	книг, выданных студентам экономического факультета
2	книг, выданных студентам определенной группы
3	студентов, получивших книги толщиной от 300 до 600 стр.
4	групп, студенты которых читают книги, изданные до 2000 г.

5	факультетов, студенты которых получили книги по математике
6	фамилий студентов, которые имеют на руках книги толщиной более 500 стр.
7	студентов, получивших на руки дорогие книги (ценой свыше 500 р.)
8	факультетов, на которых учатся студенты, читающие книги по информатике
9	книг по информатике, выданных студентам с четными номерами читательских билетов
10	книг толщиной свыше 400 стр., выданных студентам технологического факультета
11	фамилий студентов, которым выданы книги по математике или физике
12	книг по техническим наукам, выданных студентам с номерами читательских билетов, превышающими 5
13	номеров читательских билетов студентов, имеющих на руках книги Л. Н. Гумилева
14	книг по техническим наукам, выданных студентам механического факультета
15	цен особо дорогих книг (цена которых превышает 400 р.), выданных студентам экономического факультета, с указанием фамилий студентов и их групп

Таблица 3.13

Варианты заданий для составления статического запроса 2

Вариант	Задание: составить запрос для получения списка...
1	групп, студенты которых получили книги по информатике
2	факультетов, студенты которых получили книги по техническим наукам
3	студентов, которые держат на руках книги, изданные до 2006 года
4	групп, студенты которых имеют на руках книги толщиной свыше 500 стр.
5	разделов книг, полученных студентами определенной группы
6	книг по информатике, находящихся в хранилище
7	фамилий студентов, имеющих на руках книги по техническим наукам толщиной более 400 стр.

8	факультетов, студенты которых получили книги дешевле 500 р.
9	книг малой стоимости (цена менее 300 р.), выданных студентам технологического факультета с номерами билетов, превышающими 4
10	книг по естественным наукам, находящихся в хранилище
11	книг по математике, выданных на руки (т.е. не находящихся в хранилище)
12	разделов, книги по которым не выданы на руки
13	книг по математике и техническим наукам, выданных студентам механического факультета
14	изданных до 2007 года книг по техническим наукам
15	выданных на руки книг по общественным наукам дороже 500 р. с указанием номеров билетов читателей, которые имеют их на руках

Таблица 3.14

Варианты заданий для составления параметрического запроса

Вариант	Задание: составить запрос для получения списка
1	книг, выданных студентам какого-то факультета
2	книг, выданных студентам какой-то группы
3	номеров читательских билетов студентов, имеющих на руках книги по какой-то теме
4	студентов, которым выданы книги по какому-то разделу
5	факультетов, студенты которых получили книги по некоторому разделу
6	студентов, которые имеют на руках книгу некоторой толщины
7	студентов, получивших книгу какой-либо цены
8	групп, студенты которых читают книги, изданные в каком-либо году
9	факультетов, на которых учатся студенты, читающие книги какого-либо автора
10	книг толщиной свыше 400 стр., выданных студентам какого-либо факультета
11	книг, выданных студенту с каким-либо номером читательского билета
12	книг, выданных студентам некоторого факультета с номерами читательских билетов, превышающими 4

13	студентов, получивших книги по некоторому разделу толщиной от 200 до 400 стр.
14	книг по некоторому разделу, выданных студентам какого-либо ф-та
15	групп, студенты которых получили книги по какому-либо разделу

Задания для самостоятельной работы. Часть V.

Теоретические положения

Создание формы на основе параметрического запроса

Во второй части практической работы была создана простая форма для регистрации новой книги, включавшая все поля одной таблицы – «Список книг». В определенных случаях требуются более сложные формы для изменения записей в таблицах.

В качестве примера рассмотрим ситуацию – читатель возвращает в библиотеку взятую книгу. Ее надо либо поместить в хранилище, либо выдать другому читателю, желающему ее получить. Для этого надо быстро отыскать в БД запись о книге и изменить значение поля «Читатель».

Поиск записи о книге удобно провести по ее инвентарному номеру с помощью параметрического запроса. Для редактирования записи можно использовать форму, которая предоставит для редактирования именно эту запись. Такая форма создается на основании запроса.

Сначала надо создать и сохранить параметрический запрос «Поиск книги», параметром которого служит значение поля «№». В результате выполнения запроса должны отображаться поля, перечисленные в табл. 3.15.

Таблица 3.15

Поля запроса «Поиск книги»

Таблица БД	Поле	Примечания
«Список книг»	№	Параметр поиска
«Список книг»	Авторы	Отображается для дополнительного контроля правильности выполняемых действий
«Список книг»	Название	Отображается для дополнительного контроля правильности выполняемых действий
«Список книг»	Читатель	Редактируемое поле
«Список читателей»	Ф. И. О.	Отображается для дополнительного контроля правильности выполняемых действий

Для создания формы воспользуемся Мастером форм, который запускается щелчком по кнопке «Создание» – «Формы» – [Другие формы] – [Мастер форм]. Работа с Мастером состоит в заполнении окон ввода в поочередно возникающих окнах диалога:

а) в первом окне (рис. 3.20) выбираются таблица или запрос, на основании которых создается форма (их может быть несколько), и поля, которые должны быть в ней отображены. Наша форма создается на основании запроса «Поиск книги» и в ней надо отобразить все поля, перечисленные в табл. 3.15. Это можно быстро сделать щелчком по кнопке [>>], затем надо щелкнуть [Далее];

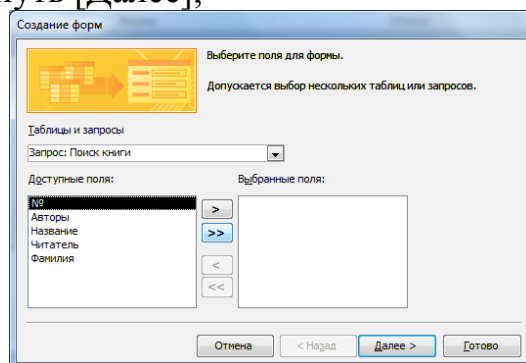


Рис. 3.20. Выбор полей для формы

б) во втором окне (рис. 3.21) выбирается внешний вид формы путем соответствующей установки переключателя. Схема в левой части окна показывает структуру, которую примет форма при том или ином выборе. Для нашего примера выберем вариант «в один столбец». Затем надо щелкнуть [Далее];

в) в третьем окне выбирается стиль формы;

г) в четвертом окне надо задать имя формы и щелкнуть [Готово]. При этом происходит сохранение формы, и она открывается для работы. Т.к. форма создана на основании параметрического запроса, при открытии формы СУБД запрашивает значение параметра – номера книги – с помощью окна, аналогичного изображенному на рис. 3.19. После ввода параметра открывается форма, структура которой аналогична рис. 3.6. Для ее редактирования надо перейти в режим Конструктора.

Изменение свойств отдельного элемента, добавление надписей, создание заголовка и примечания проводятся так же, как это делалось в части 2 практической работы. Изменение расположения элементов и их размеров по отдельности для формы, созданной Мастером форм, возможны, если при создании выбран внешний вид формы «выровненный» (рис. 3.21).

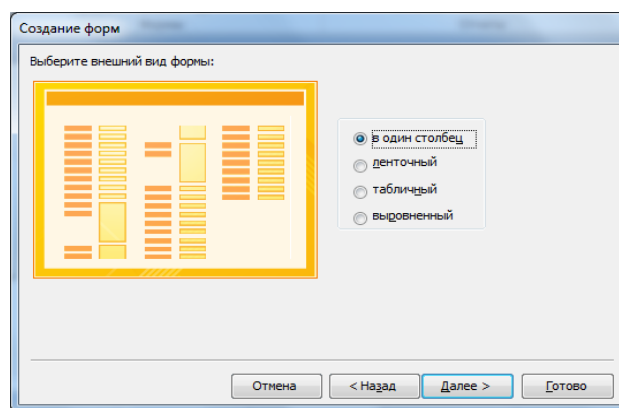


Рис. 3.21. Выбор внешнего вида формы

Возможные варианты компоновки формы показаны на рис. 3.22.

Поскольку форма предназначена только для перерегистрации книги, т.е. изменения значения поля «Читатель», значения остальных полей надо защитить от случайных изменений. Для этого при установлении свойств соответствующих окон ввода надо на вкладке «Данные» страницы свойств (рис. 3.8) установить значение свойства «Доступ» – «Нет». После этого установка курсора в окно и изменение значения поля станут невозможными.

Задания для самостоятельной работы

1. Создать с помощью Мастера форм форму для перерегистрации книги по рассмотренному примеру (вид формы – «выровненный»).

2. В режиме Конструктора произвести оформление формы: а) кегль шрифта во всех элементах формы выбрать равным 14, прочие параметры оформления для полей, описывающих книгу («№», «Авторы», «Название») и их подписей должны быть такими же, как для надписей в задании к части 2 (табл. 3.8), а для полей, описывающих читателя («Читатель», «Ф.И.О.») и их подписей – такими же, как для окон ввода в том же задании. Изменить расположение элементов формы для вариантов с нечетными номерами – по образцу рис. 3.22а; для вариантов с четными номерами – рис. 3.22б. Размеры надписей и полей подобрать самостоятельно;

б) изменить текст подписи поля с «Читатель» на «№ читательского билета»;

в) добавить в область данных формы надпись-инструкцию: «Изменить только № читательского билета!». Обеспечить неизменность всех полей, кроме поля «Читатель»;

г) озаглавить форму «Перерегистрация книги»;

д) создать примечание формы, в котором указать свою фамилию в качестве разработчика.

3. С помощью формы произвести перерегистрацию какой-либо книги.

а) б)
Рис. 3.22. Варианты компоновки формы

Задания для самостоятельной работы. Часть VI.

Теоретические положения

Создание отчета

Таблицы, получаемые в результате выполнения запроса, являются временными. Результат выполнения запроса можно сохранить, создав по нему отчет.

Допустим, мы хотим иметь сведения о том, студенты каких факультетов интересуются книгами отдельных тематических разделов. Для этого сначала создадим и сохраним запрос, в котором будут указаны: тематический раздел, параметры книги (авторы, название, место и год издания) и факультет, на котором учится читатель книги. Книги, находящиеся в хранилище, в результаты запроса попасть не должны.

Структура запроса показана на рис. 3.23. К перечисленным выше полям добавлено поле «Читатель» из таблицы «Список книг». Оно не отображается в результирующей таблице (снят флажок «Вывод на экран») и служит для создания условия отбора, по которому в результат не попадают книги «читателя № 1», т.е. находящиеся в хранилище.

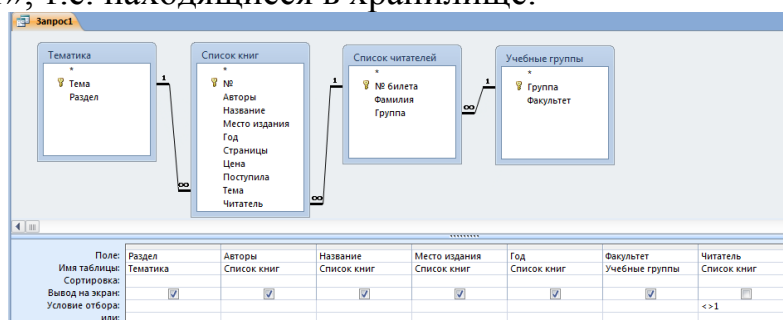


Рис.3.23. Запрос для создания отчета

После этого надо щелкнуть кнопку «Создание» – «Отчеты» – [Мастер отчетов]. Как и в других случаях, работа с Мастером заключается в последовательном выборе параметров отчета с помощью ряда окон диалога. После выбора в каждом окне надо щелкнуть [Готово] для перехода к следующему окну.

а) в первом окне (его вид аналогичен рис. 3.20) выбираются основа для создания отчета (только что созданный и сохраненный запрос) поля для отображения в отчете. Следует выбрать для отчета все поля, перечисленные в окне (т.к. поле «Читатель» не отображается в результате запроса, оно не предлагается и для занесения в отчет);

б) во втором окне (рис. 3.24) выбирается вид представления данных – с группировкой или без нее. Слева в окне перечислены таблицы, использованные в запросе. Вид представления данных зависит от выбора таблицы.

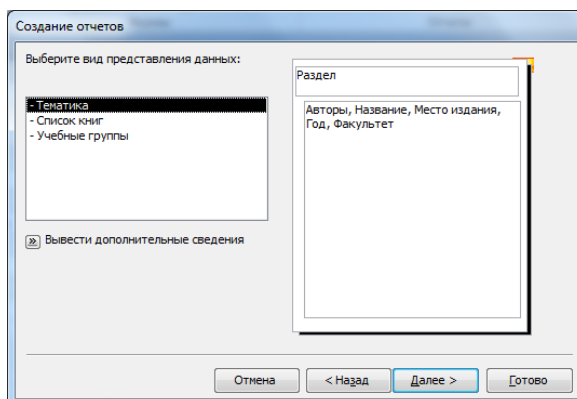


Рис. 3.24. Выбор представления данных в отчете

Если выбрана таблица, из которой в запрос вошло только одно поле («Тематика» или «Учебные группы»), то выбранные записи будут автоматически собраны в группы, соответствующие отдельным значениям этого поля. При выборе таблицы, из которой в запрос вошло несколько полей, группировка не производится. Структура групп отображается в правой части окна.

В нашем случае записи следует сгруппировать по разделам, поэтому слева выбирается таблица «Тематика».

в) в третьем окне при необходимости можно задать следующий уровень группировки. В нашем примере этого делать не надо;

г) в четвертом окне задается способ сортировки записей внутри группы – выбирается поле и вид сортировки, аналогично тому, как это делается в таблицах Word или Excel.

Щелкнув в этом окне клавишу [Итоги], можно для каждой группы задать вид подводимых итогов для имеющихся числовых полей (в нашем отчете это поле «Год»). Мастер отчетов предлагает четыре простейших вида итогов – сумму (Sum), среднее арифметическое (Avg), максимальное или минимальное значение имеющегося в таблице числового поля («Год»). Задать подведение более сложных итогов можно в режиме Конструктора.

Установим флажок в позиции «Min», чтобы оценить возраст самой старой выданной читателю книги по каждому разделу.

д) в пятом окне выбирается макет отчета. Это делается с помощью переключателя. В левой части окна схематически отображается структура размещения элементов будущего отчета. Выберем по умолчанию вариант «ступенчатый»;

е) в шестом окне можно выбрать стиль оформления отчета. В левой части окна отображаются параметры оформления, присущие выбранному стилю;

ж) в седьмом, последнем, окне задается имя отчета, под которым он будет сохранен в БД. В этом окне надо щелкнуть [Готово].

Вид полученного отчета показан на рис. 3.25.

Раздел	Авторы	Название	Место издания	Год	Факультет
Естественные	Артемюк А.И.	Органическая химия	М.: Высшая шк	2005	Эконом
	Касаткина Н.В.	Физическая химия	М.: РИОР	2007	Техноло
	Кузнецов В.В.	Физиология растений	М.: Высшая шк	2005	Эконом
Итого для 'Раздел' = Естественные науки (3 записей)				2005	
Информатика	Харрингтон Д.	Проектирование реляционн	М.: Лори	2006	Механи
Итого для 'Раздел' = Информатика (1 запись)				2006	
Математика	Зельдович Я.Б.	Элементы прикладной матем	СПб.: Лань	2002	Эконом

Рис. 3.25. Отчет, полученный с помощью Мастера

Редактирование отчетов режиме Конструктора

Анализируя рис. 3.25, можно видеть, что структура отчета, созданная Мастером, не слишком удачна. Для полей «Название» и «Факультет» не хватает места. В то же время столбец для поля «Авторы» слишком широк. Для изменения структуры надо войти в режим Конструктора (рис. 3.26). Это делается через контекстное меню, вызываемое правым щелчком по закладке отчета.

Раздел	Авторы	Название	Место издания	Год	Факультет
Раздел					

Рис. 3.26. Конструктор отчетов

В окне Конструктора видны семь рабочих областей, границы которых можно перетаскивать мышью:

а) область заголовка отчета, текст которого можно отредактировать, установив в окно курсор;

б) область верхнего колонтитула, содержащая надписи – заголовки полей, отображаемые в отчете так, как они видны в окне Конструктора (ср. рис. 3.25);

в) область заголовка группы. В отчете для каждой группы эта область содержит имя раздела (рис. 3.25);

г) область данных. Ее структура определяет вид и расположение содержимого каждой из групп, соответствующих определенному разделу;

д) область примечания группы, в которой помещаются итоги. Как видно из рис. 3.26, содержимое пояснения к итогам формируется с помощью выражения строкового типа

="Итоги для " & " 'Раздел' = " & " " & [Раздел] & " (" & **Count(*)** &
" " & **If(Count(*)=1;"запись";"записей")** & ")"

Это выражение состоит из отдельных текстовых фрагментов, соединенных оператором сцепления **&**. Название поля в квадратных скобках: [Раздел] – в отчете заменяется на название поля «Раздел» для соответствующей группы (см. рис. 3.25). Кроме того, в выражение входят две функции (выделены полужирным курсивом):

– функция **Count(*)** подсчитывает количество записей в группе;

– функция **If(Count(*)=1; "запись"; "записей")** аналогична функции **ЕСЛИ(...)** электронных таблиц Excel. Если условие **Count(*)=1**, которое является первым аргументом функции, выполнено, функция возвращает значение, являющееся ее вторым аргументом – слово запись. В противном случае возвращается слово записей. Это сделано для того, чтобы числовое значение и грамматическая форма слова «запись» более или менее соответствовали друг другу, однако в отчете все равно появляется грамматически неправильное сочетание «3 записей» (рис. 3.25).

Во второй строке этой области помещаются подпись «**Min**» и значение итога, вычисляемое с помощью одноименной функции: **=Min([Год])**.

е) область нижнего колонтитула страницы, в которой по умолчанию помещаются текущая дата и номер страницы отчета (на рис. 3.25 эта область не видна);

ж) область примечания отчета. По умолчанию она свернута (отсутствует). Чтобы ее создать, надо мышью оттащить вниз ее границу.

Принципы работы с Конструктором отчетов аналогичны принципам работы с Конструктором форм. Элементы отчета могут удаляться, форматироваться и т.д., при этом окно какого-либо поля в области данных связано с окном подписи к этому полю в области верхнего колонтитула. Изменять расположение и размеры связанных элементов отчета независимо

друг от друга при создании его с помощью Мастера невозможно. Сделанное в окне Конструктора изменение относится ко всем элементам соответствующего вида в отчете.

Например, можно проделать следующие действия:

- в области заголовка раздвинуть окно надписи и изменить ее текст на «Тематика выданных книг»;
- в области верхнего колонтитула изменить размеры и расположение окон заголовков полей (это отразится и на размерах полей в области данных);
- в области заголовка группы отодвинуть вниз границу области и увеличить вертикальный размер окна имени раздела, чтобы имя могло выводиться в две строки (см. рис. 3.27);
- в области данных изменить кегль шрифта в отдельных окнах так, чтобы содержимое было полностью видно в окне;
- в области примечания изменить текст примечания на
="Самый ранний год издания по разделу" & " " & [Раз-дел] & " (" & "записей: " & **Count(*)** & "): "
- имеющий более понятный смысл в рамках данного отчета (см. рис. 3.27). Кроме того, удалить окно надписи "Min", а окно значения итога поместить в ту же строку, что и текст примечания;
- создать примечание отчета с указанием имени разработчика;
- изменить гарнитуру, кегль, цвет и начертание символов в окнах и некоторые другие параметры оформления.

В результате отчет принимает вид, показанный на рис. 3.27.

Тематика выданных книг					
Раздел	Авторы	Название	Место издания	Год	Факультет
Естественные науки	Кузнецов В.В.	Физиология растений	М.: Высшая школа	2005	Технологический
	Кузнецов В.В.	Физиология растений	М.: Высшая школа	2005	Технологический
	Теллер Е.З.	Практикум по микробиологии	М.: Дрофа	2004	Технологический
	Теллер Е.З.	Практикум по микробиологии	М.: Дрофа	2004	Механический
Самый ранний год издания по разделу Естественные науки (записей: 4):				2004	
Информатика	Харрингтон Д.	Проектирование реляционных баз данных	М.: Лори	2006	Технологический
а) верхняя часть страницы					
Самый ранний год издания по разделу Технические науки (записей: 3):				2004	
Физика	Агеев Е.П.	Неравновесная термодинамика	М.: МЦНМО	2005	Технологический
	Карышев А.К.	Теплофизика	М.: МГТУ	2002	Механический
Самый ранний год издания по разделу Физика (записей: 2):				2002	
Разработчик <u>Иванов И.И., гр. ПМ-00</u>					
1 сентября 2013 г.					
Стр. 1 из 1					

б) нижняя часть страницы

Рис. 3.27. Отредактированный отчет

Задания для самостоятельной работы

1. Создать с помощью Мастера отчет на основе запроса, согласно индивидуальному варианту задания (табл. 3.16).

2. Отредактировать отчет в режиме Конструктора:

а) Выбрать для элементов отчета в отдельных областях параметры шрифта и цвет фона элементов согласно индивидуальному варианту (табл. 3.17);

б) подобрать самостоятельно ширину и кегль шрифта элементов, позволяющий полностью разместить в элементе его содержимое;

в) создать примечание, в котором указать свою фамилию в качестве разработчика.

3. Сохранить отчет.

Таблица 3.16

Варианты отчета
(сортировку вести по возрастанию, если не указано иначе)

№	Поля запроса	Поле группировки	Поля сортировки	Поле и вид итога	Макет оформления отчета
1	Авторы, Название, Поступила, Цена, Группа, Тема	Тема	Поступила, убывание	Цена, сумма	Ступенчатый
2	Название, Место издания, Год, Факультет, Группа	Факультет	Группа, затем – Год (убывание)	Год, максимум	Блок
3	Авторы, Название, Цена, Раздел, Группа	Раздел	Цена	Цена, среднее	Структура
4	№, Раздел, Название, Страницы, Фамилия, Факультет	Факультет	Фамилия	Страницы, максимум	Ступенчатый
5	№, Раздел, Название, Страницы, Фамилия, Факультет	Раздел	№	Страницы, минимум	Блок
6	Тема, Авторы, Страницы, Год издания, Цена, Группа	Тема	Цена, убывание	Год, максимум	Структура

Продолжение табл. 3.16

7	Группа, Авторы, Название, Место издания, Год, Страницы	Группа	Год издания	Страницы, среднее	Ступенчатый
8	Тема, Название, Цена, Поступила, Факультет	Факультет	Поступила	Цена, сумма	Блок
9	Авторы, Название, Поступила, Страницы, Группа, Фамилия	Группа	Фамилия	Страницы, среднее	Структура
10	Авторы, Название, Тема, Цена, Поступила, Факультет	Тема	Факультет	Цена, максимум	Ступенчатый
11	№, Раздел, Название, Цена, Место издания, Группа, Фамилия	Раздел	№	Цена, минимум	Блок
12	№, Цена, Год, Место издания, Тема, Факультет	Факультет	Цена, убывание	Цена, среднее	Структура
13	Название, Цена, Поступила, Группа	Группа	Цена	Цена, сумма	Ступенчатый
14	Раздел, Группа, Фамилия, Авторы, На-звание, Цена	Раздел	Группа, затем фамилия	Цена, максимум	Блок
15	№, Раздел, Название, Страницы, Фамилия, Факультет	Раздел	Страницы	Страницы, минимум	Блок

Таблица 3.17

Параметры форматирования отчета

№	Заголовок	Верхний колонтитул	Заголовок группы	Область данных	Примечание отчета
1	Выравнивание по центру области, шрифт Times New Roman, черный, фон – белый	Шрифт Courier New, полужирный курсивный белый, фон – синий. Выравнивание в окнах – по левому краю	Шрифт Arial желтый, фон – голубой	Шрифт Cambria курсивный коричневый, фон – белый. Выравнивание в окнах – по центру	Шрифт Calibri полужирный зеленый, фон – белый
2	Выравнивание по левому краю области, шрифт Courier New курсивный, синий, фон – белый	Шрифт Arial желтый, фон – голубой. Выравнивание в окнах – по центру	Шрифт Cambria курсивный коричневый, фон – белый	Шрифт Calibri полужирный зеленый, фон – белый. Выравнивание в окнах – по левому краю	Шрифт Times New Roman полужирный курсивный красный, фон – светло-серый
3	Выравнивание по центру области, шрифт Arial полужирный, белый, фон – синий	Шрифт Cambria курсивный коричневый, фон – белый. выравнивание в окнах – по левому краю	Шрифт Calibri полужирный зеленый, фон – белый	Шрифт Times New Roman полужирный курсивный красный, фон – светло-серый. Выравнивание в окнах – по центру	Шрифт Courier New синий, фон – красный

Продолжение табл. 3.17

№	Заголовок	Верхний колонтитул	Заголовок группы	Область данных	Примечание отчета
4	Выравнивание по левому краю области, шрифт Cambria полужирный курсивный, желтый, фон – голубой	Шрифт Calibri полужирный зеленый, фон – белый. Выравнивание в окнах – по центру	Шрифт Times New Roman полужирный курсивный красный, фон – светло-серый	Шрифт Courier New синий, фон – красный. Выравнивание в окнах – по левому краю	Шрифт Arial курсивный черный, фон – светло-серый
5	Выравнивание по центру области, шрифт Calibri, коричневый, фон – белый	Шрифт Times New Roman полужирный курсивный красный, фон – светло-серый. Выравнивание в окнах – по левому краю	Шрифт Courier New синий, фон – красный	Шрифт Arial курсивный черный, фон – светло-серый. Выравнивание в окнах – по центру	Шрифт Cambria полужирный коричневый, фон – голубой
6	Выравнивание по левому краю области, шрифт Times New Roman курсивный, зеленый, фон – белый	Шрифт Courier New синий, фон – красный. Выравнивание в окнах – по центру	Шрифт Arial курсивный черный, фон – светло-серый	Шрифт Cambria полужирный коричневый, фон – голубой. Выравнивание в окнах – по левому краю	Шрифт Calibri полужирный курсивный желтый, фон – черный

7	Выравнивание по центру области, шрифт Courier New полужирный, красный, фон – светло-серый	Шрифт Arial курсивный черный, фон – светло-серый. Выравнивание в окнах – по левому краю	Шрифт Cambria полужирный коричневый фон – голубой	Шрифт Calibri полужирный курсивный желтый, фон – черный. Выравнивание в окнах – по центру	Шрифт Times New Roman белый, фон – синий
8	Выравнивание по левому краю области, шрифт Arial полужирный курсивный, синий фон – красный	Шрифт Cambria полужирный коричневый фон – голубой. Выравнивание в окнах – по центру.	Шрифт Calibri полужирный курсивный желтый, фон – черный	Шрифт Times New Roman белый, фон – синий. Выравнивание в окнах – по левому краю.	Шрифт Courier New курсивный синий, фон – желтый
9	Выравнивание по центру области, шрифт Cambria черный, фон – светло-серый	Шрифт Calibri полужирный курсивный желтый, фон – черный. Выравнивание в окнах – по левому краю	Шрифт Times New Roman белый, фон – синий	Шрифт Courier New курсивный синий, фон – желтый. Выравнивание в окнах – по центру	Шрифт Arial полужирный черный, фон – светло-зеленый
10	Выравнивание по Calibri курсивный,	Шрифт Times New Roman белый, фон – синий. Выравнивание в окнах – по центру	Шрифт Courier New курсивный синий, фон – желтый	Полужирный черный, зеленый. Выравнивание по левому краю	Шрифт Cambria полужирный курсивный зеленый, фон – голубой

Продолжение табл. 3.17

№	Заголовок	Верхний колонтитул	Заголовок группы	Область данных	Примечание отчета
11	Выравнивание по центру области, шрифт Times New Roman полужирный, желтый, фон – черный	Шрифт Courier New курсивный синий, фон – желтый Выравнивание в окнах – по левому краю	Шрифт Arial полужирный черный, фон – светло-зеленый.	Шрифт Cambria полужирный курсивный зеленый, фон – голубой. Выравнивание в окнах – по центру	Шрифт Calibri белый, фон – темно-зеленый
12	Выравнивание по левому краю области, шрифт Courier New полужирный курсивный, белый, фон – синий	Шрифт Arial полужирный черный, фон – светло-зеленый. Выравнивание в окнах – по центру	Шрифт Cambria полужирный курсивный зеленый, фон – голубой	Шрифт Calibri белый, фон – темно-зеленый. Выравнивание в окнах – по левому краю	Шрифт Times New Roman курсивный синий, фон – белый
13	Выравнивание по центру области, шрифт Arial синий, фон – желтый	Шрифт Cambria полужирный курсивный зеленый, фон – голубой. Выравнивание в окнах – по левому краю	Шрифт Calibri белый, фон – темно-зеленый	Шрифт Times New Roman курсивный синий, фон – белый. Выравнивание в окнах – по центру	Шрифт Courier New, полужирный курсивный белый, фон – синий

14	Выравнивание по левому краю области, шрифт Cambria курсивный, черный, фон – светло-зеленый	Шрифт Calibri белый, фон – темно-зеленый. Выравнивание в окнах – по центру	Шрифт Times New Roman курсивный синий, фон – белый	Шрифт Courier New, полужирный курсивный белый, фон – синий. Выравнивание в окнах – по левому краю	Шрифт Arial желтый, фон – голубой
15	Выравнивание по центру области, шрифт Calibri полужирный, зеленый, фон – голубой	Шрифт Times New Roman курсивный синий, фон – белый. Выравнивание в окнах – по левому краю	Шрифт Courier New, полужирный курсивный белый, фон – синий	Шрифт Arial желтый, фон – голубой. Выравнивание в окнах – по центру	Шрифт Cambria курсивный коричневый, фон – белый