

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Государственное образовательное учреждение
высшего профессионального образования
МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«МАМИ»

В.И.Калядин, А.И.Макаров

Основы работы на персональном компьютере
Сборник лабораторных работ
по дисциплине «Информатика» для студентов всех специальностей

МОСКВА
2010

УДК 681.3.06

Разработано в соответствии с Государственным образовательным стандартом 2008 г. для всех специальностей на основе примерной программы дисциплины «Информатика»

Рецензенты: к.э.н., доц. кафедры «Информационные технологии в экономике» П.А.Аркатов;

к.т.н., с.н.с. ФГУП «ИНИСТИ им. акад. А.И.Берга»
В.Ф.Блохина.

Работа подготовлена на кафедре "Информационные системы и дистанционные технологии"

Калядин В.И., Макаров А.И. Основы работы на персональном компьютере. Сборник лабораторных работ по дисциплине «Информатика» для студентов всех специальностей. М.: МАМИ, 2010. 84 с.: ил.

Сборник ориентирован на освоение приемов работы с операционной системой Windows и изучение основ обработки информации с помощью программ Word, Access, Excel и макросов на VBA.

Введение и лабораторные работы №4, 5, 6 написаны В.И.Калядиным, лабораторные работы №1, 2, 3 написаны А.И.Макаровым.

©Калядин В.И., Макаров А.И., 2010.

©МГТУ «МАМИ», 2010.

Введение

Настоящий сборник лабораторных работ ориентирован на освоение студентами приёмов работы с широко распространённой операционной системой MS Windows и приложениями пакета Microsoft Office. В сборнике рассмотрены следующие вопросы:

- работа с файловой системой с помощью программ Проводник и Командная строка, подготовка простейших текстов и документов в редакторах Блокнот и WordPad, простейшие вычисления на Калькуляторах;
- подготовка документов в текстовом процессоре Word;
- подготовка таблиц с данными, вычисления по этим данным, построение графиков в MS Excel;
- создание баз данных и работа с ними в СУБД MS Access;
- обработка в Excel табличных данных с использованием макросов на VBA.

Для выполнения лабораторных работ студенту следует:

- изучить теоретический материал лабораторной работы по лекциям, методическим указаниям и рекомендуемой литературе;
- ознакомиться с заданием к лабораторной работе и подготовить в отчете материал для её выполнения.

Отчет по лабораторной работе должен быть оформлен на листах бумаги формата А4 и включать в себя следующие разделы:

- титульный лист с названием лабораторной работы;
- задание к лабораторной работе;
- материал для выполнения лабораторной работы в соответствии со своим конкретным вариантом задания.

На занятиях студенту следует:

- показать преподавателю оформленный отчет с материалами для выполнения лабораторной работы и получить допуск к работе на компьютере,
- выполнить на компьютере задание лабораторной работы,
- внести полученные результаты в отчет,
- представить результаты выполненной работы и отчет преподавателю для защиты. Защитить лабораторную работу, отвечая на предложенные преподавателем вопросы и выполняя контрольные задания.

Лабораторная работа №1

Операционная система Windows.

1. Краткие теоретические сведения

Операционная система Windows реализована на базе объектно-ориентированной концепции разработки программных средств. При использовании такого подхода к программированию центральное место занимает объект. С каждым объектом связывают два основополагающих понятия - его свойства и методы, позволяющие изменять первые.

На уровне пользователя объектный подход выражается в том, что программный интерфейс представляет собой подобие реального мира, а работа с машиной сводится к действиям с привычными объектами. Так, например, документы хранятся в папке, ее можно открыть, документы - просмотреть, исправить, переложить с одного места на другое или выбросить в корзину и т.п.

В рамках этого подхода центральным объектом операционной системы являются окна, через которые организуется диалог с пользователем. При этом в системе «стандартизованы» не только внешний вид окон, но и средства управления ими. Такое единство подхода существенно упрощает работу пользователя не только с операционной системой, но и с программами (в терминах Windows - приложениями).

Файловая система хранения информации

Вся информация на долговременных носителях данных компьютера хранится в файлах. *Файл* - именованный набор однотипных данных. Это может быть текстовый документ, рисунок, программа и т.п. *Имя* файла состоит из двух частей - *собственно имени* и (через точку) *расширения*. Длина имени файла до 255 символов. Имя файла может быть любым и задаётся пользователем, а расширение, как правило, устанавливается автоматически, так как оно связывает файл с создавшей его программой или предназначенной для его обработки.

Каталог - это элемент структуры организации файлов на диске, когда группа файлов объединяется под единым именем по соображениям удобства пользователя. Каталог может содержать не только фай-

лы, но и другие каталоги (подкаталоги). В Windows термин каталог заменен более понятным в бытовом смысле названием *папка*.

Любой файл однозначно определяется компьютером с помощью своего *полного имени*, которое включает в себя имя накопителя, где хранится файл, *имена вышележащих каталогов* и собственно имя. Например, полное имя текстового файла с именем *Fio.txt*, хранящегося в папке *Иванов*, вложенной в папку *1АА-2* на диске *I:*, имеет вид: *I:/1АА-2/Иванов/Fio.txt*.

Рабочий стол Windows

Пользовательский интерфейс операционной системы Windows спроектирован так, чтобы создать максимально удобные условия для пользователя. Значительное внимание уделено тому, чтобы пользователь концентрировал свое внимание в первую очередь на объектах и документах, а не на программах, с помощью которых они создаются и обрабатываются.

В силу этого интерфейс Windows представляет собой подобие обычного рабочего стола, на котором располагаются значки папок с документами, принтера, самого компьютера, корзины для выбрасываемых документов и то, что может понадобиться для работы пользователя.

После загрузки системы на экране появляется *Рабочий стол*, на котором размещены различные графические значки (иконки) отображающие доступные пользователю объекты (рис.1.1).



Рис. 1.1 Пример рабочего стола Windows

Пользователь может изменить внешний вид, расположение, размеры и цвета отдельных элементов или экрана в целом. После установки операционной системы Windows на компьютер на *Рабочем столе* располагаются следующие графические значки (иконки).

Мой компьютер позволяет получить доступ к ресурсам компьютера. При его активизации (двойной щелчок) на экран выводится окно, содержащее значки дисков, привода компакт-дисков, сетевых каталогов, к которым подключен компьютер, и *Панели управления*.

Корзина - используется для временного сохранения удаленных файлов.

В нижней части *Рабочего стола* располагается *Панель задач* - основное средство взаимодействия с системой. На ней изображена кнопка с логотипом Windows и надписью *Пуск*. Она обеспечивает открытие программ и документов, а также доступ к объектам, которые уже были открыты

Щелчок левой клавишей мыши по кнопке *Пуск* приводит к появлению на экране *Главного меню* операционной системы. Главное меню позволяет выполнить быстрый доступ ко всем программам, установленным в системе. Выход из системы также осуществляется с помощью *Главного меню*.

Щелчок правой клавишей мыши по кнопке *Пуск* приводит к появлению на экране контекстного меню, одна из позиций которого позволяет запустить файловый менеджер Windows – программу *Проводник*.

В центральной части *Панели задач* (справа от кнопки *Пуск*) располагается панель инструментов *Быстрый запуск*, на которую вынесены кнопки запуска часто используемых программ. Правее на *Панели задач* располагаются кнопки, каждая из которых соответствует работающей в данный момент программе или открытой папке. Кнопка имеет имя приложения или папки, что позволяет переключаться с одной задачи на другую путем их нажатия.

Справа на *Панели задач* выводятся индикатор текущей раскладки клавиатуры, часы и некоторые другие. *Индикатор клавиатуры* может использоваться для переключения клавиатуры на разные языки. Буквы *En* означают, что будут вводиться буквы из английского алфавита, буквы *Ru* на индикаторе - русского. Щелчок левой кнопкой мыши по

Индикатору клавиатуры вызывает окно, позволяющее переключиться с одного языка на другой.

Структура окна

Центральным объектом операционной системы является *окно* - прямоугольная область на экране монитора. Именно в окнах осуществляется работа с программами и папками. Все элементы оконного интерфейса стандартизированы. Некоторые окна могут иметь в своем составе дополнительные части и элементы, однако, наиболее общая структура окна представлена на рис. 1.2.

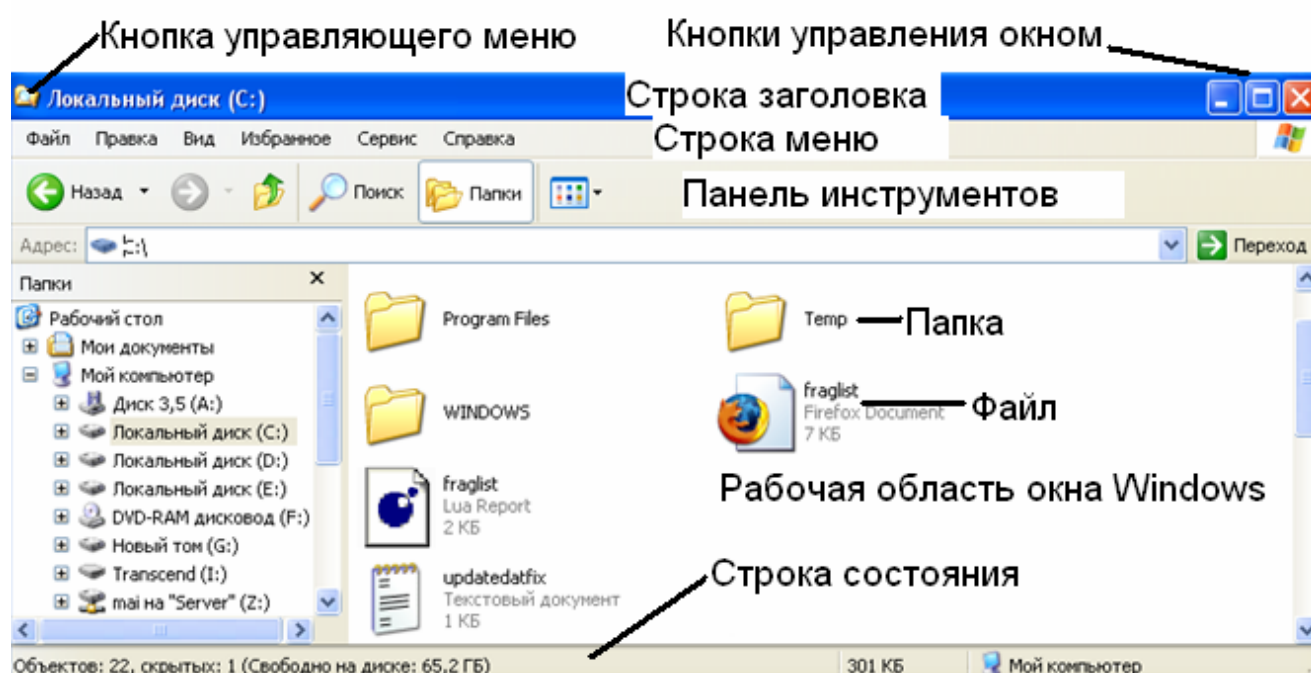




Рис. 1.2. Рабочее окно программы *Проводник* Windows

Ниже перечислены основные части окна и их назначение.

Кнопка управляющего меню. Представляет собой значок в левом углу заголовка. Он открывает меню содержащее команды по изменению размера и положения окна на экране.

Кнопки управления окном. Три кнопки в правой части заголовка окна:

Кнопка свертывания - сворачивает окно и помещает его на *Панель задач*;

Кнопка разворачивания  - переводит окно в *полноэкранный режим*. При этом изображение кнопки заменяется на *Кнопку свернуть в окно* , которая служит для восстановления первоначального размера окна;

Кнопка закрытия - завершает работу программы и закрывает окно.

Строка заголовка. Верхняя строка окна, где указывается имя прикладной программы или открытого документа.

Строка меню. Вторая строка окна, в которой располагаются заголовки меню с основными командами управления программой.

Панель инструментов. Панель, располагается в третьей строке окна, и в некоторых приложениях может занимать несколько строк. На панели находятся кнопки быстрого доступа к командам. Вид панели (перечень предлагаемых инструментов) меняется в зависимости от программы или функции окна.

Рабочая область - содержательная часть окна. Например, при работе файлового менеджера в рабочей области располагаются иконки папок и файлов, а при работе с документом в рабочей области появляется его текст.

Строка состояния - строка, в которой выводится информация об объекте или выбранных командах.

Программа Проводник

Для работы с файлами или папками (запуск программ, копирование, переименование или удаление файлов и папок) в Windows используется программа Проводник. Для ее запуска необходимо выполнить цепочку команд *Программы-Стандартные-Проводник*, либо из контекстного меню кнопки *Пуск* (рис. 1.2). Окно Проводника содержит по умолчанию две области: слева - область структуры папок, называемую *Папки*, и справа - область, в которой отображается содержимое выбранной папки.

Создание папки

Чтобы создать папку, требуется выделить в окне Проводника папку, в которой будет размещена новая, и выполнить цепочку команд

Файл-Создать-Папку, либо через контекстное меню, вызываемое щелчком правой кнопки мыши. В правой области Проводника появляется значок папки с именем по умолчанию *Новая папка*. Вместо слов «Новая папка» необходимо ввести в их поле новое имя создаваемой папки и нажать клавишу *Enter*.

Управление файлами и папками

Хранящиеся на диске файлы и папки можно переносить, копировать, удалять. Прежде чем выполнить эти операции, объекты необходимо выделить. Для выбора одного объекта достаточно установить на него курсор и щелкнуть кнопкой мыши - он выделится другим цветом. Для выделения нескольких объектов, используются следующие приемы.

- ✓ Обвести курсором группу рядом стоящих объектов прямоугольной рамкой, держа нажатой левую кнопку мыши.
- ✓ Выделить первый объект и, удерживая клавишу *Shift*, выделить остальные объекты.
- ✓ Чтобы выделить несколько объектов, не расположенных рядом, необходимо выделить первый из них, нажать клавишу *Ctrl* и, удерживая ее, выделить поочередно все остальные.
- ✓ Для выделения в папке всех объектов следует нажать *Ctrl+A*.

Чтобы отменить выделения достаточно щелкнуть мышью, в любое место окна или нажать любую клавишу управления курсором.

Копирование и перемещение объектов

Копирование и перемещение файлов и папок выполняется одинаково. Следует помнить, что при копировании и перемещении файла копируется или перемещается только этот файл, а при копировании или перемещении папки — все находящиеся в ней файлы.

Копирование и перемещение объектов можно выполнить через, так называемый, *Буфер Обмена*, представляющий собой область памяти, доступную всем приложениям Windows, либо через механизм «*перенести-и-оставить*». При работе с *Буфером Обмена* выделенные объекты сначала копируются в *Буфер Обмена* командами *Копировать*

или *Вырезать*, а затем, копируются или перемещаются в нужную папку по команде *Вставить*.

Механизм «перенести-и-оставить»

Для визуального перемещения объекта или группы объектов в окне Проводника с помощью мыши надо выполнить следующие действия:

- ✓ выделить в правой области окна Проводника перемещаемые объекты любым из описанных выше способов;
- ✓ добиться, чтобы в левой области Проводника отображалась папка, в которую будут перенесены объекты;
- ✓ установить указатель мыши на выделенные объекты, нажать левую кнопку мыши и, удерживая ее, переместить указатель в левую область окна на значок папки, в которую перемещаются объекты;
- ✓ отпустить кнопку мыши.

Чтобы скопировать объект или группу объектов требуется выполнить ту же операцию, но при перетаскивании объектов в окно папок надо удерживать нажатой клавишу *Ctrl*.

Операции копирования и переноса объектов можно также осуществлять правой кнопкой мыши. После отпускания кнопки на экран будет выведено меню, где следует выбрать – перемещать объекты или копировать.

Если при выполнении копирования или перемещения объектов допущена ошибка, то для возврата к исходному состоянию следует выполнить команду *Отменить* из команд панели инструментов.

Текстовый редактор «Блокнот»

В поставку Windows входит простейший текстовый редактор *Блокнот*. В *Блокноте* можно подготовить текстовые файлы, имеющие расширение ТХТ. Запускается Блокнот по цепочке команд *Пуск-Программы-Стандартные-Блокнот*.

Окно программы имеет строку заголовка, строку меню и рабочую область. Текст документа вводится абзацами. Завершается набивка абзаца нажатием клавиши *Enter*. При вводе информации каждый аб-

защита вводится как строка бесконечной длины, что затрудняет просмотр текста. Команда *Перенос по словам* в меню *Правка* позволяет ограничить длину строк размерами окна. При включенной команде около нее появляется галочка.

Единственным средством форматирования документа в программе *Блокнот* является возможность выбора шрифта (его гарнитуры, начертания и размера) по команде *Шрифт* меню *Правка*.

При работе с файлами используют команды меню *Файл*: *Создать*, *Открыть*, *Сохранить*, которые позволяют создать новый файл, открыть или сохранить существующий. Команда *Сохранить Как* используется в ситуациях, когда информация записывается в новый файл, не имеющий своего имени, и при перезаписи отредактированного старого текста в новый, ранее несуществующий файл. При записи нового документа в окне *Папки* необходимо установить имя своей личной папки. Затем ввести имя создаваемого файла и нажать кнопку *Сохранить*.

Для объединения текста двух файлов с помощью программы *Блокнот* можно предложить следующую последовательность действий:

- открыть оба текстовых файла в программе *Блокнот*;
- скопировать текст второго документа в *Буфер обмена*;
- вставить скопированный текст в файл первого документа;
- сохранить первый файл под другим именем.

Командный режим системы Windows

Помимо графического режима система Windows, как и другие операционные системы, имеет командный режим. Многие возможности системы доступны только через команды. Например, с помощью команды *Tracert* <имя сервера> можно узнать, через какие узлы сети Интернет происходит связь с указанным сервером. Выход в командный режим осуществляется командой *Выполнить* Главного меню Windows. Затем в диалоговом окне *Запуск программы* следует набрать команду *cmd*. Открывается окно командного процессора, который воспринимает текстовые команды, имеющие следующий синтаксис:

Имя команды [*имя файла*] [*ключи*] [*>файл вывода*].

В качестве разделителя между параметрами команды используется один или несколько пробелов. Параметры, заданные в квадратных скобках, можно не указывать. Если не указан *файл вывода*, то информация выводится на экран монитора. Список основных команд пакетного режима можно получить по команде *Help*, а выполнив *Help команда*, можно получить сведения по конкретной *команде*. Объяснения некоторых команд, используемых в лабораторной работе, приведены ниже:

- > *имя диска*: - переход в корневой каталог указанного диска (I:);
- > *CD имя папки* – переход в папку с указанным именем (после правильного выполнения команды в подсказке командного процессора выводится имя указанной папки). Например, после команды
CD 1AA-1\Иванов подсказка командного процессора примет вид
I:\1AA-1\Иванов>;
- > *Tree* – осуществляет графическое отображение на экран или файл вывода структуры папок, подчиненных текущему каталогу (ключ */F* указывает на необходимость вывода также имен файлов, хранящихся в отображаемых папках);
- > *Dir* – выводит оглавление папки в полном формате и не отсортированном виде;
- > *Type*- выводит на экран содержимое указанного файла (не рекомендуется выводить данной командой файлы не текстового формата);
- > *Copy*- копирует один, или несколько файлов в один (при объединении нескольких файлов в один имена исходных файлов следует указывать через символ «+»).

Необходимо отметить, что информация выводимая командами в дисковые файлы сохраняется в формате «DOS», который имеет кодировку символов кириллицы, отличающуюся от кодировки формата Windows. Для конвертации таких файлов из формата DOS в формат

Windows можно использовать текстовый редактор WordPad, который входит в стандартную поставку Windows и размещается в группе *Стандартные меню Программы*.

2. Задание к лабораторной работе

1. На рабочем диске I: создать папку с номером группы (если её там нет) и в ней создать папку с фамилией. (Вся дальнейшая работа выполняется только в этой папке).

2. Открыть программу *Блокнот*. Подготовить текстовый документ, содержащий вашу фамилию, имя, отчество и номер группы. Сохранить файл под именем FIO.txt в своей личной папке. Закрыть программу *Блокнот*.

3. Запустить программу *Калькулятор*.

Вычислить произведение чисел: 111111; 111111 и убедиться, что калькулятор работает правильно. Эти и все последующие выражения и результаты фиксировать во вновь созданном файле Calc.txt, записывая, например, так: «111111*111111=??».

Используя обычный и инженерный вид калькулятора вычислить результаты одного и того же выражения $2+2*2=??$. Объяснить, почему результаты разные.

Используя вид калькулятора «Программист», записать число и месяц рождения (представленные как число) в двоичной (bin) системе счисления, а год рождения в шестнадцатеричной (hex) системе счисления, например:

Дата (число) рождения = 13 = (1101)bin,

Месяц (декабрь) рождения = 12 = (1100)bin.

Вычислить $S1=11_{16}+11_8+11_2=(11)_{\text{hex}}+(11)_{\text{oct}}+(11)_{\text{bin}}=(?)_{\text{dec}}$

Вычислить $S2=111_{16}+111_8+111_2=(111)_{\text{hex}}+(111)_{\text{oct}}+(111)_{\text{bin}}=(?)_{\text{dec}}$

4. Сохранить зафиксированные результаты в файле Calc.txt.

5. Объединить файлы FIO.txt и Calc.txt в файл Sum.txt.

6. Открыть программу *Проводник*. В своей личной папке, создать папки «Папка1», «Папка2» и «Папка3». Внутри последней создать папку «Временная».

7. Скопировать в папки «Папка1» и «Папка2» соответственно файлы FIO.txt и Calc.txt. Файл Sum.txt переместить в папку «Папка3», а затем скопировать его из этой папки в папку «Временная».

8. Используя в главном меню Windows окно «Найти программы и файлы» запустить командный процессор (набить cmd и дать Enter). Затем последовательно выполнить следующие команды операционной системы:

Z:

Cd номер_группы\фамилия

Tree /F

Tree /F >f1.txt

Type f1.txt

Dir

Dir >f2.txt

Type f2.txt

Copy f1.txt+f2.txt DOS.txt

Type DOS.txt

9. Просмотреть файл DOS.txt с помощью программы *Блокнот*.

10. Запустить программу WordPad (*Программы-Стандартные-WordPad*). Открыть файл DOS.txt как «Текстовые документы MS-DOS (*.txt)» и сохранить как «Текстовый документ» под именем «ПапкиФайлы.txt».

11. Объединить файлы Sum.txt, ПапкиФайлы.txt и сохранить результат объединения в файле Zadan1.txt.

12. Предъявить результаты работы преподавателю. После одобрения результатов содержимое файла Zadan1.txt переписать в отчёт.

Лабораторная работа №2

Работа с текстовым процессором MS Word

1. Краткие теоретические сведения

Текстовый процессор (редактор) MS Word – это приложение Windows, предназначенное для создания, просмотра, модификации и печати текстовых документов. Текстовый процессор Word работает в среде Windows и может выполнять операции над текстовой и графической информацией. Благодаря таким качествам, как функциональная эластичность, простота в работе, возможность тонкой настройки, пакет приложений Office 2007, куда входит MS Word, завоевывает все больше и больше пользователей.

Редактор Word можно запустить в работу цепочкой команд *Пуск-Программы-Microsoft Word* или щелчком левой кнопки мыши на соответствующей пиктограмме Рабочего стола.

Предлагаемый по умолчанию вид рабочего окна Word 2007, показан на рис. 2.1.

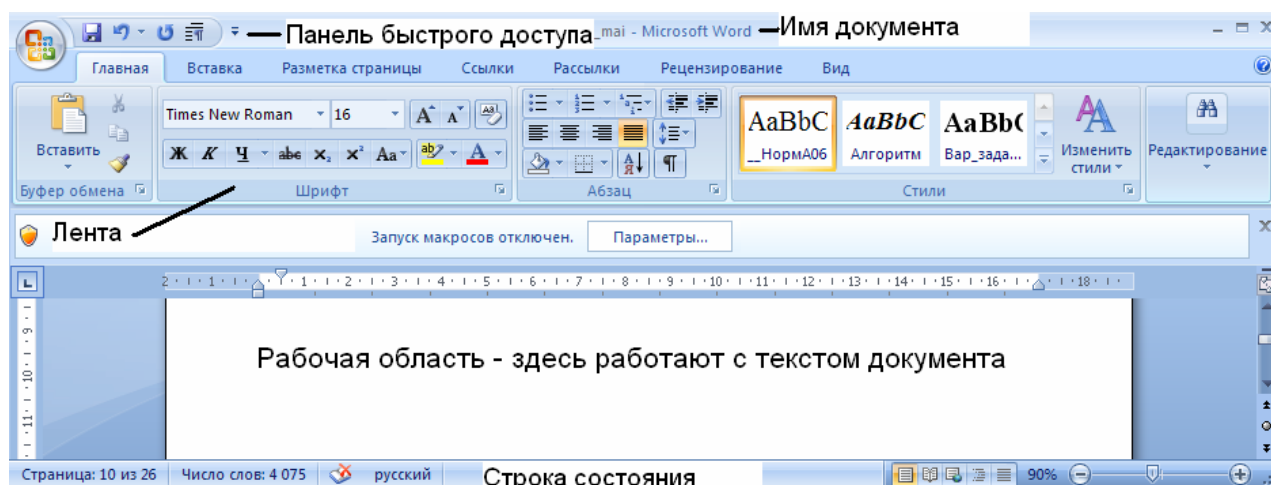


Рис. 2.1. Рабочее окно Word 2007

Основные элементы рабочего окна Word 2007

кнопка "Office" — имеет круглую форму и расположена в левом верхнем углу окна, в меню этой кнопки содержатся команды меню Файл предыдущих версий Word (Создать, Открыть, Сохранить и др.);

панель быстрого доступа — находится правее Кнопки "Office";

рабочая область — основной и самый большой элемент интерфейса, где выполняется работа с текущим документом;

лента — размещена между Панелью быстрого доступа и рабочей областью;

контекстное меню вызывается щелчком правой кнопкой мыши в рабочей области;

строка состояния — находится внизу окна и включает в себя как информационные, так и практические функции.

Панель быстрого доступа (рис. 2.2) содержит кнопки инструментов, которые используют для вызова того или иного диалогового окна либо выполнения соответствующей команды.

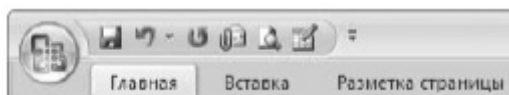


Рис. 2.2. Панель быстрого доступа

Если подвести указатель мыши к любой кнопке *Панели быстрого доступа*, то появится всплывающая подсказка с названием, а иногда — и с кратким описанием функциональности данной кнопки. Правее *Панели быстрого доступа* расположена кнопка с изображением треугольника. При ее нажатии отображается меню, в котором щелчком кнопкой мыши выбирают те команды, кнопки которых должны присутствовать на *Панели быстрого доступа*; точно так же убирают ненужные кнопки с панели.

Содержимое контекстного меню Word 2007 определяется текущим режимом работы. В частности, при работе с текстом оно содержит одни команды, с графическими объектами — совсем другие, с диаграммами — третьи и т. д. На рис. 2.3 показаны команды, которые предназначены для работы с текстом.

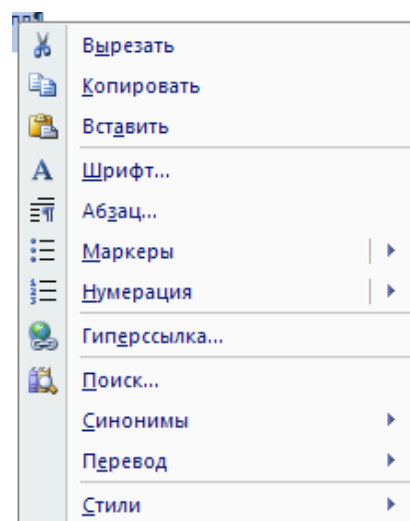


Рис. 2.3.

Первые три команды — *Вырезать*, *Копировать* и *Вставить* — необходимы для работы с текстовыми фрагментами (блоками). Они дублируют соответ-

ствующие инструменты вкладки *Главная*, которая расположена на ленте. Команда *Вырезать* предназначена для удаления выделенного текстового фрагмента из документа и запоминания его в буфере обмена. Команда *Копировать* отличается тем, что она не удаляет выделенный фрагмент из текста, а только копирует его в буфер обмена. Команда *Вставить* предназначена для вставки из буфера обмена предварительно помещенного туда текстового блока в то место, где в данный момент находится курсор. Команды *Вырезать* и *Копировать* доступны только при наличии выделенного текстового фрагмента; команда *Вставить* доступна только тогда, когда в буфере обмена содержатся какие-то данные.

С помощью команд *Шрифт* и *Абзац* переходят в режим настройки соответственно шрифта и текущего абзаца (выделенного фрагмента текста), а команды подменю *Маркеры* и *Нумерация* — создают соответственно маркированные и нумерованные списки (данные инструменты имеются также на вкладке *Главная* ленты).

Команду *Стили* используют для просмотра и изменения стилей, применяемых для оформления текста.

Команда *Поиск* включает режим поиска информации. После выполнения команды появляется окно, в котором задают условия поиска и запускают процесс поиска.

Вид ленты Word 2007, предлагаемый по умолчанию» показан на рис. 2.1.. Каждая вкладка ленты включает в себя однотипные и функционально сходные элементы управления, которые объединены в группы. В частности, вкладка *Главная* содержит группы *Буфер обмена*, *Шрифт*, *Абзац* и *Стили*. В некоторых группах, правее названия, имеется небольшого размера кнопка со стрелочкой, после нажатия которой открывается соответствующий режим работы (для одной группы — меню с командами, для другой — диалоговое окно).

Создание нового файла Word осуществляется командой *Создать Кнопки "Office"*. В результате откроется окно (рис. 2.4), в котором нужно указать режим создания нового документа. Создать документ можно либо на основе шаблона, либо без него. Под шаблоном понимается совокупность настроек, стилей и т. п., которые применяются к

документу. В частности, для деловых писем можно использовать один шаблон, для отчетов — другой. Если создается новый документ без применения шаблона, то в левой части окна *Создание документа* выбирается в поле *Шаблоны* раздел ***Пустые и последние***, затем в центральной части данного окна следует щелкнуть кнопкой мыши на значке *Новый документ* и нажать кнопку *Создать*. Откроется окно нового документа Word 2007, которому по умолчанию будет присвоено имя *Документ1*.

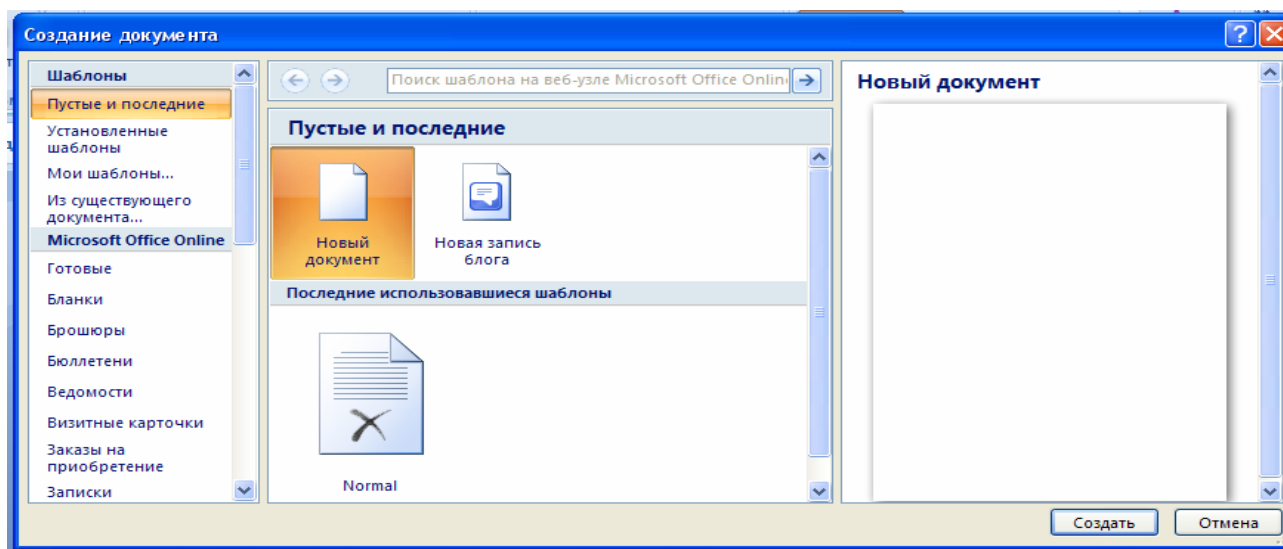


Рис. 2.4. Создание нового документа

Для открытия созданного ранее документа Word следует выполнить команду *Открыть* в меню *Кнопки "Office"* (либо нажать комбинацию клавиш *Ctrl+O*). В результате откроется окно, где следует определить папку, где лежит документ его имя и тип.

Подготовительные операции

Работа с документом начинается с выбора размера и ориентации листа бумаги. Наиболее распространен формат А4 (210x297 мм). Ориентация бывает книжная и альбомная. У документа имеются поля относительно края листа бумаги. Для оформления отчетов и рефератов используют следующие размеры полей:

Левое поле	30 мм
Правое поле	15 мм
Верхнее поле	20 мм
Нижнее поле	20 мм

Установка параметров страницы производится на вкладке *Разметка страницы* ленты.

На закладке *Поля* устанавливаются требуемые размеры полей. На закладке *Размер* выбирают размер бумаги. На закладке *Ориентация* выбирают *Книжную* или *Альбомную* ориентацию бумаги.

В многостраничном документе используют так называемые колонтитулы - области наверху или внизу каждой страницы документа. В этих областях документа проставляются номера страниц и краткие заголовки. Колонтитулы располагаются в областях верхнего и нижнего полей листа и могут использоваться как отдельно (только верхний или только нижний), так и совместно. Задание колонтитулов можно выполнить вызвав контекстное меню соответственно в верхней или нижней части страницы.

При создании документа необходимо продумать базовый стиль его оформления, то есть выбрать гарнитуру шрифта, его размер и начертание, расстояние между строками, метод выравнивания текста в абзаце, наличие отступа в первой (красной) строке и его размер, интервал между абзацами.

Установка вышеуказанных атрибутов шрифта и абзаца производится с помощью вкладки **и** ленты через опции *Шрифт* и *Абзац*.

На рис. 2.5. приведено окно Шрифт, используемое для установки параметров шрифта документа.

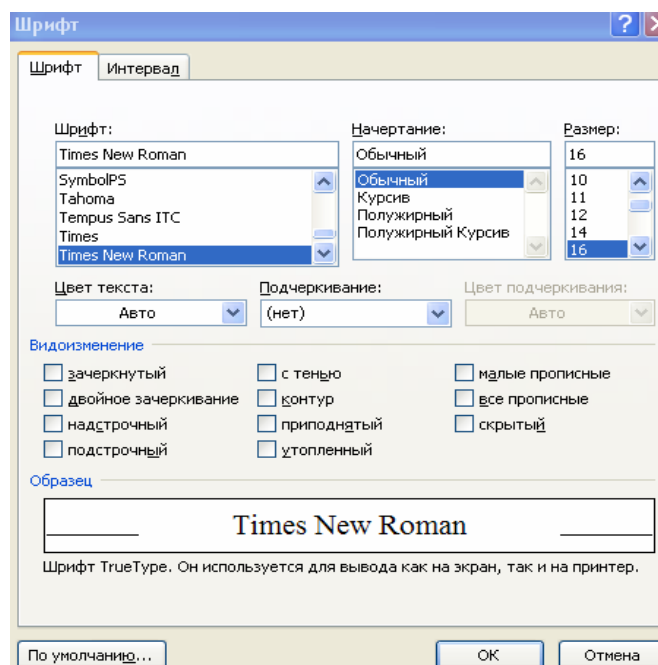


Рис.2.5. Окно *Шрифт*

Шрифты характеризуются тремя основными параметрами: гарнитурой, высотой (кегель) и начертанием (полужирный и курсив). Можно рекомендовать для основного текста шрифт Times New Roman обычного начертания. Для основного текста формата А4 принято использовать 12, 13 или 14 пунктов.

Формат, опции Абзац приведен на рис.2.6.

Отступ - это расстояние абзаца от левого или правого края страницы. Обычно используется для специального выделения абзаца из общего текста. Отдельно регулируется отступ первой строки. Этот параметр позволяет формировать «красную» строку. Величина отступа ус-

танавливается отдельно и обычно составляет 1,0 - 1,25 см.

Интервал - расстояние между абзацами. Интервал задается значениями перед и после абзаца. Особо устанавливается междустрочный интервал. Обычно его устанавливают значением *Полуторный* или *Множитель* 1,2-1,4.

При выравнивании все строки абзаца начинаются или заканчиваются в одной и той же позиции. Печатные издания принято выравнивать по ширине.

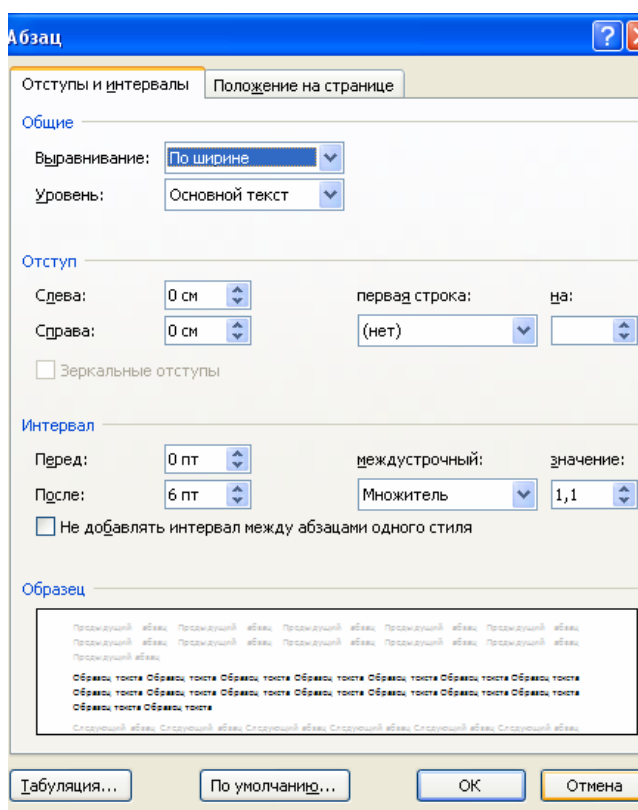


Рис.2.6. Окно *Абзац*

Ввод и редактирование текста

Ввод текста может выполняться в режиме *вставки* или *замещения*. В режиме *Вставки* вводимые по месту расположения курсора символы автоматически сдвигают находящийся справа текст. В режиме *Замещения* вводимый по месту расположения курсора текст замешает находящиеся справа от курсора символы. Переключение между режимами вставки и замещения выполняется с помощью клавиши *Ins*.

При вводе текста документа требуется помнить одно правило - абзац вводится как единая строка бесконечной длины. Разбивки его на строки в зависимости от настройки параметров документа выполняется редактором автоматически. Переход к следующему абзацу осуществляется при нажатии клавиши *Enter*.

Если требуется вставить в документ символ, отсутствующий на клавиатуре, используют команду *Символ* вкладки *Вставка* ленты.

Форматирование текста — это процедура, без которой не обходится практически ни один документ: всегда нужно как-то оформить заголовки, создать список, выделить стилем текстовый фрагмент или слово и т. д. При создании и обработке печатных документов могут использоваться следующие виды форматирования:

- Форматирование символов;
- Форматирование абзацев;
- Форматирование документа в целом.

Часто в документах встречаются фрагменты, содержащие перечисления действий или предметов. Для улучшения восприятия таких фрагментов используются *Списки*. Word позволяет формировать списки двух типов: маркированные и нумерованные. Построение списков возможно как в автоматическом режиме, так и в режиме форматирования.

В автоматическом режиме в начало абзаца вносится цифра с точкой (скобкой) - нумерованный список, или символ с пробелом - маркированный список. При нажатии клавиши *Enter* для перехода к следующему абзацу он будет автоматически получать очередной номер или указанный маркер.

В режиме форматирования необходимо выделить требуемые абзацы и выполнить команду *Маркеры* или *Нумерация* контекстного меню, либо закладки *Маркеры*, *Нумерация* или *Многоуровневый список* группы *Абзац* вкладки *Главная* ленты.

Запись математических формул.

В MS Word 2007 имеется встроенная поддержка записи и изменения формул. Для перехода в режим ввода формулы следует на вклад-

ке *Вставка* в группе *Символы* выполнить команду *Формула*. Такого же результата можно добиться нажав клавишу $\text{Ctrl}+=$. Ввод в формулу букв русского и латинского алфавитов, а также простейших арифметических операторов, выполняется с клавиатуры. Прочие символы вводятся с помощью панели инструментов выведенных взамен ленты (рис. 2.7).



Рис 2.7

Если щелкнуть по треугольнику, изображенному рядом с командой *Формула*, то выводится встроенная в Word коллекция формул, которую можно использовать как готовые шаблоны.

Однако если формула создана с помощью надстройки Microsoft Equation 3.0 в более ранней версии Microsoft Word, изменить эту формулу можно только в Equation 3.0. Для изменения формулы, созданной с помощью надстройки Equation 3.0 необходимо дважды щелкнуть формулу, которую требуется корректировать, левой кнопкой мыши и внести необходимые изменения.

Использование таблиц

Таблицы используются для наглядного и компактного представления данных. Они являются необходимым элементом научно-технической и экономической документации.

При создании таблиц следует использовать группу *Таблица* вкладки *Вставка* ленты. В результате выполнения команды *Вставить таблицу* открывается окно *Вставка таблицы*, где надо указать число строк и столбцов создаваемой таблицы, а также установить нужные переключатели *Автоподбор ширины столбцов* (рис.2.8).

Если выбрать переключатель *Автоподбор ширины столбцов - Постоянная*, то устанавливаются равные значения ширины для всех столбцов. Значения вводятся в поле рядом. При выборе переключателя *По содержимому* – ширина столбцов увеличивается в соответст-

вии с вводимым текстом. Переключатель *По ширине окна* устанавливает общую ширину таблицы равной ширине бумаги.

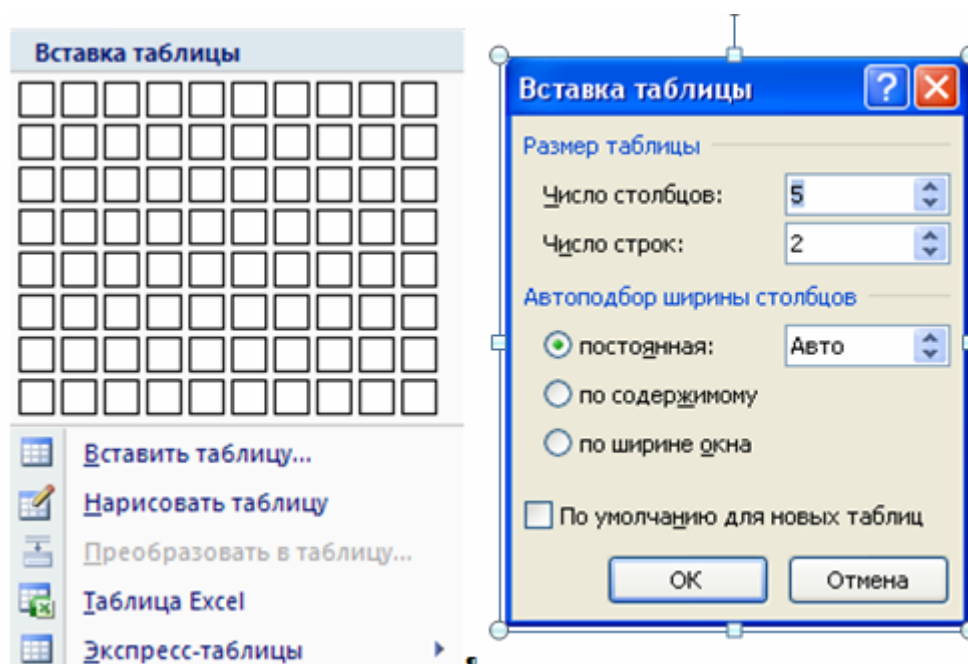


Рис. 2.8

В программу Word заложено представление о таблице, как о некотором аналоге *Базы данных*. Это позволяет выполнять с таблицей некоторые несложные операции: сортировку данных, вычисления по формулам и др.

Для сортировки столбца таблицы следует выполнить следующие действия:

1. Установить курсор в таблице.
2. В группе *Работа с таблицей* щелкнуть вкладку *Макет*, а затем в группе *Данные* выбрать команду *Сортировать*.
3. В диалоговом окне *Сортировка* выбрать необходимые параметры (рис. 2.9).
4. Для сортировки данных необходимо выбирать первый столбец, по которому будет осуществляться сортировка, и указать тип его данных. Затем, если необходимо, устанавливаются параметры вложенных столбцов (второго и третьего). Переключатель *Список* позволяет сортировать информацию таблицы со строкой заголовка или без него. Кнопка *Параметры* устанавливает язык,

используемый для сортировки, и, если необходимо, чувствительность к регистру.

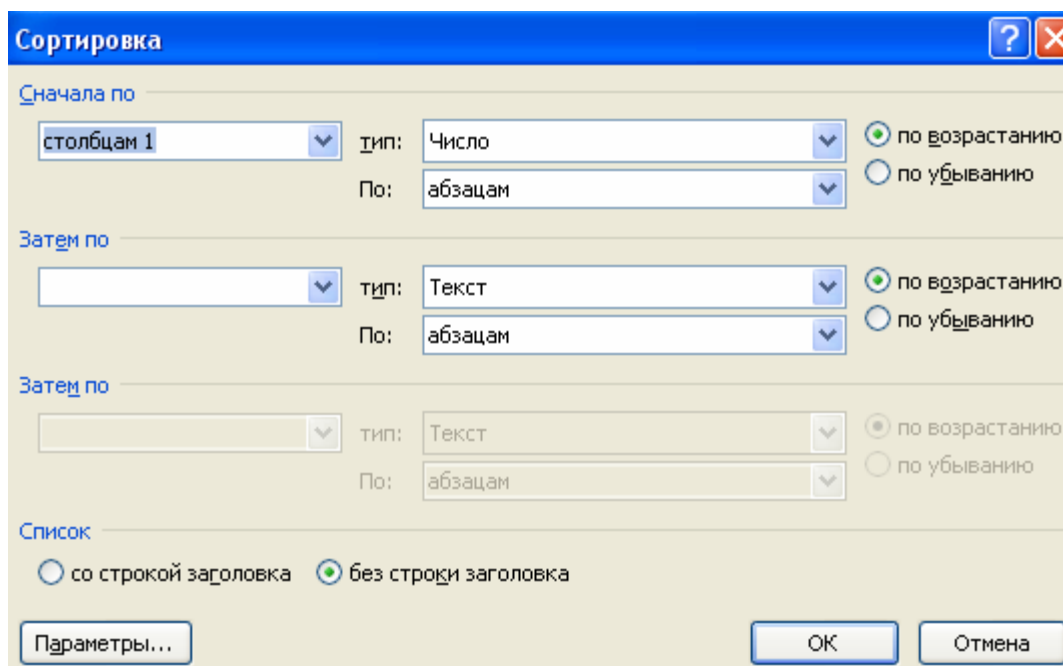


Рис.2.9

Вычисления в таблице базируются на внутренней адресации ячеек таблицы, представленной на рис.2.10.

Каждая ячейка таблицы имеет адрес, состоящий из буквы, идентифицирующей столбец, и цифры – строку. Так выделенная на рисунке ячейка имеет адрес B3.

	A	B	C	D	E
1					
2					
3					
4					
5					

Рис. 2.10.

Вычисления в таблице выполняются в следующей последовательности: выделяют ячейку, в которую должен быть помещен результат вычислений; выполняется команда *Формула...* группы *Данные макета таблицы*, в результате чего на экран выводится окно (рис.2.11).

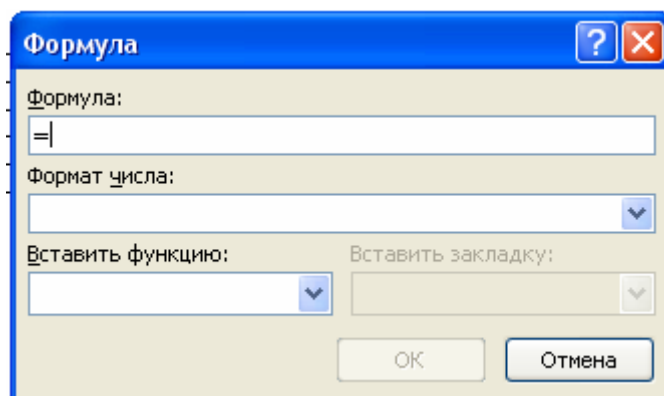


Рис.2.11

В поле окна *Формула*, после знака равенства вводится вычисляемое выражение, записанное через адреса ячеек таблицы, хранящих исходные данные. В поле *Формат числа*, если необходимо, вводится формат результата вычислений, а поле *Вставить функцию* используется для ввода в формулу математических функций, например: `sum()` – сумма; `count()` – количество; `max()` – максимум; `abs()` – модуль, `average()` – вычисление среднего значения и другие. После нажатия на кнопку *ОК* вычисляется математическое выражение, а полученный результат помещается в ячейку таблицы. Для указания диапазона ячеек при вычислении функции используются: имена ячеек (a2;a5), сверху (above) от активной ячейки, слева (left) от активной ячейки и др. Например, для вычисления среднего значения чисел в столбце можно задать функцию `average(c2:c4)`, либо `sum(above)/3`.

Графические объекты

В процессоре Word допускается использовать для оформления документа графические изображения. При этом различают *графический объект* и *рисунок* как два основных понятия, определяющие возможности и технологию применения графических средств Word.

Графический объект представляет собой единое изображение геометрической фигуры (линии, дуги, овала, прямоугольника или фигуры произвольной формы). При создании и обработке графических объектов Word допускаются: изменение их размеров и заполнение нужным цветом, задание линии контура, управление размещением на переднем плане или на фоне других фигур и текста и др.

Рисунок - сложное изображение, сформированное другими программными средствами. В Word'е поддерживается ряд возможностей по работе с рисунками, в частности, обеспечивается регулировка яркости и контрастности, можно задавать теневой или объемный эффекты.

Так как рисунок вставляется в слой текста, то для его свободного перемещения по документу необходимо выполнить команду *Обтекание текстом* контекстного меню, где необходимо изменить режим обтекания *В тексте* на любой другой и, тем самым, перевести его в графический слой.

Сохранение документа

Обязательным этапом завершающей стадии работы с документом является его сохранение в файле. Для сохранения текущего документа в состав меню *Кнопки "Office"* включены команда *Сохранить*, которая активизируется также нажатием *Shift+F12*, и *Сохранить как*. Команду *Сохранить* применяют, как правило, для сохранения изменений в текущем документе после предварительного его сохранения в файле. Команда *Сохранить Как* требует от пользователя указания папки, куда сохраняется документ, его имени и типа файла. Многие приложения пакета Office 2007 позволяют сохранять файлы в новых форматах. Например, в Word 2007 можно сохранить текстовый документ не только в привычном формате DOC, но и в DOCX, который недоступен для предыдущих версий данного текстового редактора.

2.Задание к лабораторной работе

При подготовке к работе студент должен:

1. Прочитать краткие теоретические сведения по лабораторной работе.
2. Ознакомиться с заданием для выполнения работы в редакторе Word.
3. Переписать в отчет по лабораторной работе из таблицы 2.1 параметры документа своего варианта, разбив их на 3 группы: параметры страницы, параметры шрифта, параметры абзаца.
4. Написать свою автобиографию (пример автобиографии приведен ниже).
5. Записать формулу своего варианта из таблицы 3.2 лабораторной работы 3.

6. Зарисовать схему формируемого документа с указанием имен вставляемых рисунков.
7. Показать отчет преподавателю и получить допуск к работе на компьютере.

Задание для выполнения работы в Word

1. Установить текущие значения шрифта, абзаца и страницы для своего варианта по таблице 1.
2. Напечатать автобиографию с абзацами и с перечислением в виде маркированного списка результатов вступительных испытаний в МГТУ «МАМИ». Ввести заголовок «Автобиография», в конце вставить дату и подпись.
3. Напечатать, по указанию преподавателя, 2 абзаца текста из методички и ввести заголовок «Фрагмент теоретической части лабораторной работы».
4. Напечатать в начале документа общий заголовок «Практическая часть лабораторной работы №2».
5. Опробовать работу команд: *Найти*, *Заменить* группы *Редактирование* вкладки *Главная* ленты и *Проверка орфографии* (клавиша F7).
6. Применить к общему заголовку стиль «Заголовок 1», а двум другим заголовкам стиль «Заголовок 2».
7. Установить автоматическую нумерацию заголовков, выбрав из предложенных образцов Многоуровневого списка формат «1. Заголовок1».
8. Сохранить полученный документ в своей папке - вначале под именем «Word1», а затем «Zadan2».
9. Ввести заголовок третьего раздела «Редактор формул» и применить к нему стиль «Заголовок 2». Создать средствами Word 2007 формулу из своего варианта задания к лабораторной работе №3. Ввести свою фамилию и в виде поля дату и время (вкладка *Вставка*, группа *Текст*, команда *Дата и время*). Установить режим выравнивания подписи *По правому краю*.

10. Задать разрывы страницы перед заголовками второго и третьего разделов. Ввести нумерацию страниц сверху страницы и задать верхний колонтитул «Подготовка документов в MS Word». Установить выравнивание «По левому краю» и обрамление одной тонкой линией снизу.

11. Добавить понравившиеся рисунки в виде надписи в первый и третий разделы. К рисункам добавить название.

12. На основе маркированного списка из автобиографии создать таблицу, осуществить автоматическую нумерацию строк, отсортировать таблицу по столбцу с баллами вступительных испытаний и рассчитать средний балл, полученный на вступительных испытаниях.

№	Название вступительного испытания	Полученный балл
	Математика	70
	Русский язык и литература	65
	Физика	68
Средний балл		67,67

13. Добавить перед таблицей заголовков «Таблица Word» стилем «Заголовок 2», сохранить файл, вывести на экран все страницы документа в режиме «Предварительный просмотр» и предъявить работу преподавателю.

Пример автобиографии для выполнения пункта 2 задания.

Автобиография

Я, Иванов Петр Васильевич, родился 13 февраля 1993 года в г. Видное Московской области. Мои родители: мать – Иванова Марина Сергеевна, 1972 года рождения, бухгалтер; отец – Иванов Василий Петрович, 1968 года рождения, автомеханик.

В 2010 году окончил СОШ № 368 ВАО г. Москвы. В период учебы в школе выполнял общественную работу: был редактором школьной газеты, капитаном команды КВН. Занимался в секции классической борьбы, имею 1 юношеский разряд.

В 2010 году поступил в МГТУ «МАМИ» на специальность «Автомобиле- и тракторостроение» со следующими результатами вступительных испытаний:

- математика 70 баллов;
- русский язык и литература 65 баллов;
- физика 68 баллов.

В настоящее время учусь в группе 1АА-2 факультета АТФ.

01.09.2010 П.Иванов

Таблица 2.1.

№ вар.	Левая граница [мм]	Правая грани- ца	Верхняя грани- ца	Нижняя грани- ца	Ширина страницы	Высота страницы	Отступ [мм]	Междустроч- ный	Размер шрифта	Тип шрифта
1.	25	10	25	20	20	25	11	1.4	13	Times
2.	23	15	24	19	20,1	25,2	12	1.3	10	Arial
3.	29	14	23	18	20,2	25,4	12,5	1.2	14	Courier
4.	25	13	22	17	20,3	25,6	13	1.1	12	Verdana
5.	17	12	21	16	20,4	25,8	15	1.0	13	Garamond
6.	25	11	20	15	20,5	26	10	1.5	14	Tahoma
7.	30	10	19	14	20,6	25,8	12	1.4	12	Tunga
8.	25	11	18	13	20,7	25,6	12,5	1.3	13	Book Antica
9.	30	12	17	12	20,8	25,4	13	1.3	14	Comic Sans
10.	25	13	16	11	20,9	25,2	15	1.1	11	Tahoma
11.	20	14	15	10	21	25	10	1.5	10	Tunga
12.	15	15	16	11	20	25,2	12	1.4	12	Arial
13.	30	16	17	12	20,1	25,4	12,5	1.3	11	Courier
14.	25	17	18	13	20,2	25,6	13	1.2	13	Verdana
15.	20	18	19	14	20,3	25,8	15	1.0	11	Garamond
16.	15	19	20	15	20,4	26	10	1.5	14	Times
17.	20	20	21	16	20,5	25,8	12	1.3	11	Arial
18.	25	19	22	17	20,6	25,6	12,5	1.1	13	Courier
19.	30	18	23	18	20,7	25,4	15	1.5	11	Verdana
20.	25	17	24	19	20,8	25,2	10	1.3	14	Comic Sans
21.	20	16	25	20	20,9	25	12,5	1.1	12	Book Antica
22.	15	15	24	21	21	25,2	10	1.4	13	Times
23.	30	14	22	22	20	25,4	12	1.2	10	Arial
24.	25	13	20	23	20,1	25,6	12,5	1.0	14	Courier
25.	20	14	18	24	20,2	25,8	13	1.5	12	Verdana
26.	25	15	16	25	20,3	26	15	1.0	13	Garamond
27.	30	16	20	20	20,4	25,8	10	1.3	14	Tahoma
28.	19	17	24	15	20,5	25,6	12	1.4	12	Tunga
29.	21	18	28	10	20,6	25,4	12,5	1.2	13	Book Antica
30.	23	19	30	15	20,7	25,2	13	1.1	14	Comic Sans

Лабораторная работа №3

Работа с табличным процессором Microsoft Excel 2007

1. Краткие теоретические сведения

Программа Excel 2007 позволяет формировать и выводить на печать документы, представленные в табличном виде, выполнять расчеты на основании исходных данных и др.

Принципиально структура интерфейса Excel 2007 мало отличается от Word 2007: в ней также присутствуют *Панель быстрого доступа*, лента с вкладками, *Кнопка "Office"* и др. Меню *Кнопки "Office"* включает в себя почти те же команды, что и *Кнопка "Office"* в Word 2007, которые предназначены для создания нового документа, открытия существующего, сохранения изменений в документе либо сохранения его под другим именем, печати документа и др.

Свои особенности по сравнению с Word 2007 имеют команда *Сохранить* и подменю *Сохранить как*, что обусловлено спецификой Excel 2007. Можно сохранять документы в самых разных форматах. По умолчанию Excel 2007 предлагает сохранить документ в формате **XLSX** (этот формат прежние версии программы не понимают), но при необходимости можно выбрать тип файла, совместимый с предыдущими версиями (**XLS**). Вы можете также сохранить документ в формате **XLSM**, который поддерживает макросы.

Контекстное меню Excel 2007 в общем случае включает в себя следующие команды: *Вырезать*, *Копировать*, *Вставить* (из буфера обмена), *Специальная вставка*, *Вставить...*, *Удалить*, *Очистить содержимое*, *Вставить примечание*, *Формат ячеек*. *Выбрать из раскрывающегося списка* и *Гиперссылка*. Далее кратко рассмотрим некоторые из них.

Команда *Специальная вставка* предназначена для добавления в документ объектов из других приложений Windows. Команда *Вставить...* предназначена для добавления в документ ячеек, строк либо столбцов. При выполнении данной команды открывается окно *Добавление ячеек*, в котором указывают, какой именно элемент требуется добавить: ячейку, строку или столбец. Команда *Удалить...* работает с

точностью до наоборот — в окне *Удаление ячеек* необходимо указать, какой элемент следует удалить.

Командой *Очистить содержимое* можно при необходимости оперативно удалить содержимое активной ячейки либо группы предварительно выделенных ячеек.

Структура документа Excel

Рабочая область Excel 2007 называется элементом пользовательского интерфейса, включающим в себя ячейки, строки, столбцы и листы текущей книги, а также строку формул. Все основные действия по созданию и редактированию документов Excel 2007 выполняют именно в рабочей области.

Одним из главных структурных элементов рабочей области является ячейка. В данном случае ячейка — это наименьшая (элементарная) часть электронной таблицы, предназначенная для ввода и хранения информации и расположенная на пересечении строки и столбца. Совокупность нескольких ячеек образует диапазон.

Каждая ячейка имеет уникальные координаты (их еще называют адресом ячейки). Адрес ячейки определяется с помощью строк с буквами и цифрами, которые расположены соответственно вверху и слева рабочей области. Например, на рис. 3.1 курсор установлен в ячейку с адресом C4.

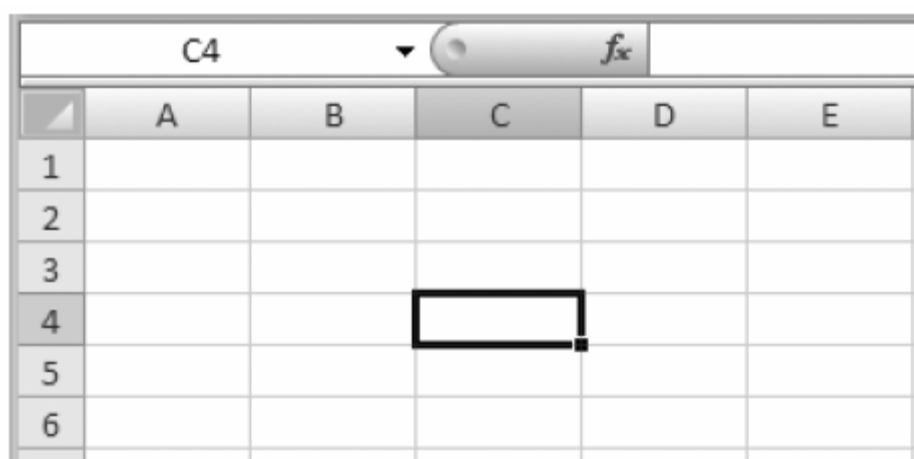


Рис. 3.1. Ячейка с координатами C4

Информацию в ячейках можно представлять в разных форматах — числовом, текстовом и т. д. Формат представления данных выбирают

в окне *Формат ячеек*, которое вызывается одноименной командой контекстного меню. Данное окно можно также открыть следующим образом: на вкладке *Главная* ленты Excel 2007 в панели *Ячейки* нужно нажать кнопку *Формат* и в появившемся меню выполнить команду *Формат ячеек*.

Рабочая книга Excel 2007 — это файл, который создается, редактируется и сохраняется средствами Excel 2007. Основной структурной единицей рабочей книги является рабочий лист.


Рабочий лист — основной элемент рабочей книги, предназначенный для ввода, редактирования и хранения данных, а также для выполнения вычислений. Основной структурной единицей рабочего листа является ячейка.

По умолчанию в состав каждой книги Excel 2007 включено три рабочих листа, которым изначально присвоены названия Лист1, Лист2 и Лист3. Однако при необходимости можно изменить количество листов книги. Как и ячейки, строки и столбцы относятся к главным составляющим рабочего листа Excel 2007. Названия столбцов по умолчанию обозначаются буквами английского алфавита, а каждая строка имеет уникальный номер. Любую строку или любой столбец можно по своему усмотрению добавлять, удалять либо перемещать, причем одновременно со всей содержащейся информацией. Что касается высоты строки и ширины столбца, то их также можно менять.

Можно использовать настройку, при которой столбцы, как и строки, будут обозначаться не буквами, а цифрами. Для этого в настройках программы в разделе формулы следует установить флажок *Стиль ссылок R1C1*, который по умолчанию снят.

Механизм автозаполнения

В программе реализована возможность быстрого ввода данных в ячейки с помощью механизма автозаполнения. Наиболее простой способ выглядит так: ввести значение в ячейку, подвести указатель мыши к правому нижнему углу ячейки (этот угол помечен жирной точкой), чтобы указатель принял вид небольшого крестика. Затем, нажав и удерживая кнопку мыши, перетащить указатель в том направлении, в котором необходимо заполнить ячейки. Как только от-

пускается кнопка мыши, все ячейки выделенного диапазона будут заполнены (в них будет скопировано значение, введенное в первую ячейку). Кроме этого, возле последней ячейки выделенного диапазона отобразится кнопка , после нажатия на которую, открывается меню, изображенное на рис. 3.2.

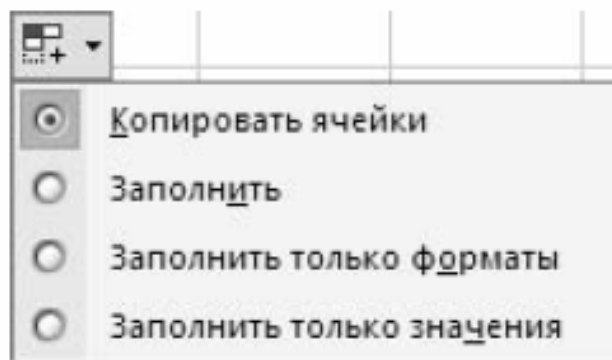



Рис. 3.2

В данном меню можно указать подходящий вариант автозаполнения ячеек. При выборе варианта *Копировать ячейки* содержимое первой ячейки выделенного диапазона будет скопировано во все остальные ячейки. Если указывается вариант *Заполнить* то все ячейки будут заполнены числовой последовательностью в порядке возрастания либо месяцами или днями недели, если в первой ячейке был введен такой текст. Например, если в первую ячейку диапазона введено значение 1, то при выборе варианта *Заполнить* значения следующих ячеек сформируются так: 2, 3, 4, 5 и т. д. При установленном значении *Заполнить только форматы* во все ячейки диапазона будет скопирован только формат первой ячейки, а при выборе варианта *Заполнить только значения* — только значение первой ячейки (форматы останутся прежними).

При заполнении ячеек числами арифметической прогрессии с шагом отличающимся от 1, следует использовать либо диалоговое окно *Прогрессия* команды *Заполнить* (значок ) группы *Редактирование* вкладки *Главная* ленты (рис. 3.3.), либо использовать следующий

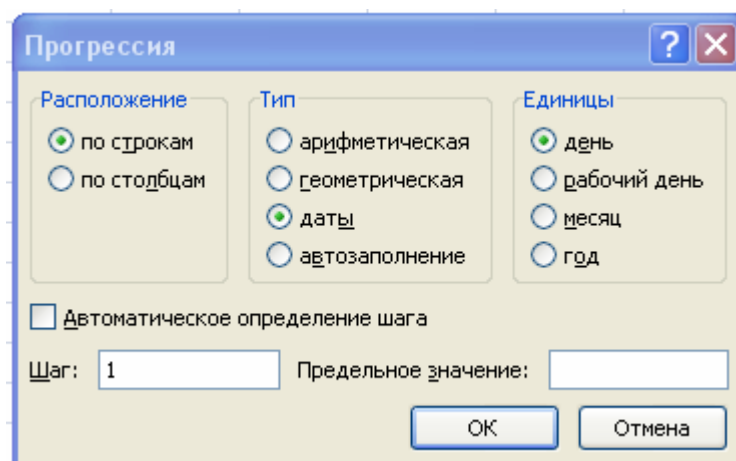


Рис. 3.3

прием. В первой ячейке набрать первый член прогрессии, в соседней ячейке — второй член прогрессии. Выделить эти ячейки и перетащить

указатель мыши в виде крестика до нужной ячейки. Заполнение ячеек числами геометрической прогрессии можно осуществить только через диалоговое окно *Прогрессия* команды *Заполнить*.

Вычисления в Excel

Строкой формул называется специальная строка, расположенная над заголовками столбцов и предназначенная для ввода и редактирования формул и иной информации. Фрагмент строки формул представлен на рис. 3.4.



Рис. 3.4. Строка формул

Строка формул состоит из двух основных частей: адресной строки, которая расположена слева, и строки ввода и отображения информации. На рис. 3.4 в адресной строке отображается значение B5, а в строке ввода и отображения информации — формула $= (A4+E7)$.

Адресная строка предназначена для отображения адреса выделенной ячейки либо диапазона ячеек, а также для ввода с клавиатуры требуемых адресов. Однако при выделении группы ячеек в адресной строке будет показан только адрес первой ячейки диапазона, расположенной в его левом верхнем углу.

В табличном редакторе Excel 2007 можно полностью автоматизировать выполнение расчетов, используя для этого формулы. Формула — это специальный инструмент Excel 2007, предназначенный для расчетов, вычислений и анализа данных.

Формула начинается со знака «=», после чего следуют операнды и операторы. Список арифметических операторов приведен в таблице 3.1. Старшинство операций при вычислении формул Excel следующее:

- операторы связи (выполняется в первую очередь);
- оператор процент;
- унарный минус;

- оператор возведение в степень;
- операторы умножение и деление;
- операторы сложение и вычитание (в последнюю очередь).

Таблица 3.1

Оператор	Действие	Пример
<i>Арифметические операторы</i>		
+	сложение	= A1 + B1
-	вычитание	= A1 – A2
*	умножение	= A2*B2
/	деление	= A2 / B2
%	взять процент	= 20%
^	возведение в степень	= A2^3
<i>Операторы связи</i>		
:	задание диапазона	=Сумма(A1:B10)
;	объединение	=Сумма(A1;A3)

Скобки в формулах Excel выполняют привычную, с точки зрения алгебры роль указания приоритета той или иной операции в выражении. Например:

$= 10*3+4^2$ дает результат 46

$$= 10*(3+4^2) \quad 190$$

Особенно внимательно надо расставлять скобки при задании унарного минуса. Например: $= -10^2$ дает результат 100, а $=(10^2)$ дает результат -100, -1^2+1^2 дает результат 2, а 1^2-1^2 дает результат 0.

Многие виды расчетов можно также выполнять с помощью специальных встроенных в Excel 2007 функций. Функция — это изначально созданная и заложенная в программу Excel процедура, которая выполняет в определенном порядке вычисления по заданным аргументам.

В состав каждой функции в обязательном порядке входят следующие элементы: имя или название (примеры имен — СУММ, СРЗНАЧ, СЧЕТ, МАКС и т. д.), а также аргумент (либо несколько аргументов), который задается в круглых скобках. Аргументами функций могут быть числа, ссылки, формулы, текст, логические величины и др.

Вводить функции можно как в ручном, так и в автоматическом режиме. В последнем случае используют мастер функций, открываемый кнопкой *Вставить функцию*, которая расположена на ленте Excel 2007 на вкладке *Формулы*.

Все имеющиеся в программе функции для удобства работы сгруппированы в категории. Выбор категории осуществляется из раскрывающегося списка *Категория*, при этом в нижней части окна отображается перечень функций, входящих в эту категорию. Если выделить требуемую функцию и нажать кнопку ОК, то откроется окно (его содержимое зависит от конкретной функции), в котором указываются аргументы функции.

Относительные и абсолютные адреса ячеек

Для записи в формулы Excel констант следует использовать абсолютную адресацию ячеек. В этом случае при копировании формулы в другую ячейку адрес ячейки с константой не изменится. Чтобы изменить в формуле относительный адрес ячейки B2 на абсолютный \$B\$2, следует нажать клавишу F4, либо вручную добавить символы доллара. Существуют также смешанные адреса ячеек B\$2 и \$B2, которые появляются после следующих нажатий на клавишу F4. Использование смешанного адреса приводит к тому, что при копировании формулы не меняется только зафиксированная часть адреса.

При копировании формулы в соседнюю правую ячейку в относительном адресе ссылки меняется буквенная составляющая. Например, ссылка C4 заменится ссылкой D4, а смешанный адрес \$C4 при копировании вдоль строки не изменится. Соответственно, при копировании формулы в соседнюю нижнюю ячейку в относительном адресе ссылки меняется цифровая составляющая. Например, ссылка C4 заменится ссылкой C5, а смешанный адрес C\$4 при копировании вдоль столбца не изменится.

Построение диаграмм

В Excel под термином диаграмма понимается любое графическое представление числовых данных. Построение диаграмм производится на основе ряда данных – группы ячеек с данными в пределах одной

строки или столбца. На одной диаграмме можно отображать несколько рядов данных.

Наиболее простой способ построения диаграмм следующий: выделить один или несколько рядов данных, в группе *Диаграммы* вкладки *Вставка* ленты Excel выбрать нужный тип диаграммы. Диаграмма будет внедрена на текущий лист рабочей книги. При необходимости ее можно перенести на другой лист с помощью команды *Переместить диаграмму* вкладки *Конструктор Работы с диаграммами*. С помощью вкладки *Макет Работы с диаграммами* можно изменить внешний вид диаграммы: добавить название диаграммы, осей, изменить шрифты и т.п.. При необходимости можно изменить подписи на горизонтальной оси. Для этого в контекстном меню диаграммы следует выбрать команду *Выбрать данные* и в диалоговом окне *Выбор источника данных* (рис. 3.5) изменить подписи горизонтальной оси.

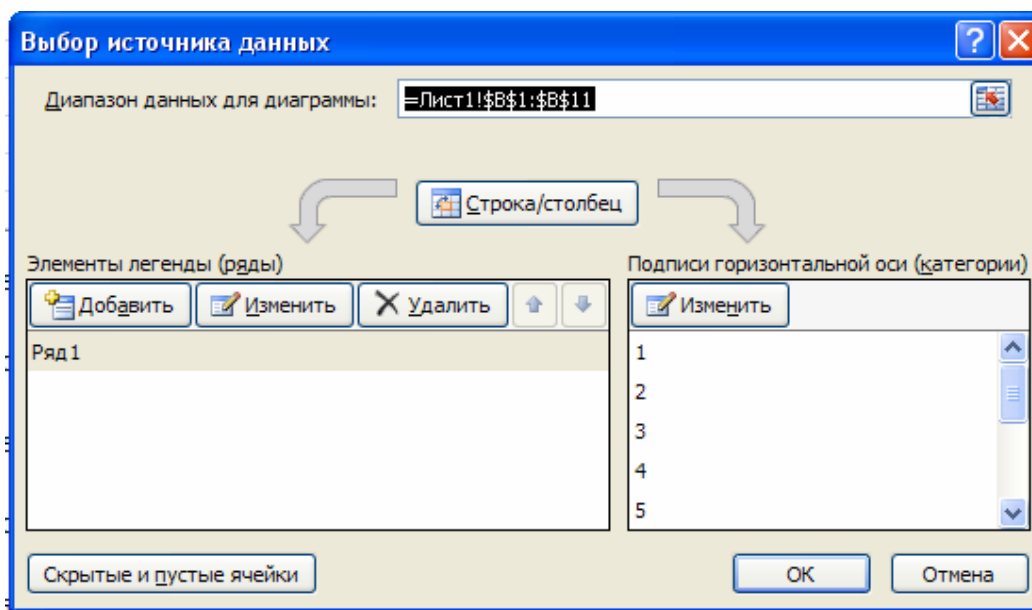


Рис. 3.5

2.Задание к лабораторной работе

Подготовка отчета к лабораторной работе

1. Прочитать краткие теоретические сведения по лабораторной работе.
2. Ознакомиться с заданием для выполнения работы в Excel. Спланировать для своего варианта структуру листов 1,2,3 и зарисовать структуру в отчет.
3. Переписать в отчет функцию из таблицы 3.2 для своего варианта.

4. Записать формулу Excel для функции одного аргумента, используя относительные адреса ячеек.
5. Записать формулу Excel для функции двух аргументов, используя смешанные адреса ячеек.
6. Показать отчет преподавателю и получить допуск к работе на компьютере.

Задание по работе в Excel

1. Переименовать «Лист1» в «Автозаполнение».

Создать во второй строке арифметическую прогрессию от 0 до 10 с разностью (шагом) 1 и в четвертой строке геометрическую прогрессию от 1 до 1024 со знаменателем 2. В первой и третьей строке ввести соответствующие заголовки.

Используя манипулятор «Мышь», задать: в первом столбце арифметическую прогрессию от 1 до 12 с шагом 1, в четвертом столбце арифметическую прогрессию от 1 до 11 с шагом 2, в седьмом столбце арифметическую прогрессию от 2 до 12 с шагом 2. В соседних столбцах соответственно задать: все месяцы года, нечетные месяцы и четные месяцы. Ввести заголовки (см. рис.3.6).

2. Построение графика функции одной переменной.

Переименовать «Лист2» в «Функция одной переменной». Построить (для своего варианта) график функции по значениям одного аргумента (см. рис. 3.7).

3. Построение графика функции двух переменных.

Переименовать «Лист3» в «Функция двух переменных». Записать в ячейку формулу со смешанным адресом. Автоматически заполнить (для своего варианта) таблицу значениями функции. Построить графики функций (см. рис. 3.8). Для проверки исправить первое значение второго аргумента на его последнее значение и убедиться, что первый и последний графики совпали.

4. Сохранить рабочую книгу в файле Zadan3.xlsx. Перерисовать полученные графики в отчет и предъявить работу преподавателю.

Таблица 3.2.

№	Функция	Значение аргумента			Значение а
		$x_{нач}$	$x_{кон}$	Δx	
1.	$y = (1/a) \cdot \exp(-(x/a)^2)$	-1,5;	1,5;	0.1	1, 1.1; 1.2, 1.3
2.	$y = x \cdot e^{-x/a}$	0;	4;	0.2	0.25; 0.5; 0.75, 1.
3.	$y = 1/\sqrt{(1-x^2)^2 + 4a^2x^2}$	0;	2;	0.05	0.1; 0.2; 0.3, 0.4
4.	$y = x \cdot \operatorname{tg} a - x^2/\cos^2 a$	0;	0,5;	0.02	15^0 ; 30^0 ; 45^0 , 60^0
5.	$y = a \cdot x^a \cdot e^{-x/a}$	0;	10;	0.25	1; 1.25; ... 2.0
6.	$y = e^{-xa} \cdot \sin x$	0;	π ;	$\pi/36$	0; 0.5; ... 2.0
7.	$y = ((x+a)^{2/3} - (x-a)^{2/3})/a$	-4;	4;	0.2	1; 2; 3, 4
8.	$y = a^{-x} - a^{-a \cdot x}$	0;	2;	0.05	10; 8; 6, 4, 2
9.	$y = (1 - \exp(-(x/a)^2))$	-2;	2;	0.1	1.25; 1.5; 1.75, 2
10.	$y = a^3/(a^2 + x^2)$	-2;	2;	0.05	0.5; 1.0; 1.5, 2.0
11.	$y = \frac{\sqrt[4]{(a+1)x + e^{-x^3}}}{\sqrt{2ax}}$	1;	7;	0,25	0.5; 0.75; 1.0, 1.25
12.	$y = \frac{\sqrt[3]{ax^2 + e^{-x^2}}}{\sqrt{(a+1)x}}$	1;	7;	0,25	0.5; 0.75; 1.0, 1.25
13.	$y = \frac{\sqrt[3]{x + ae^{-x^2}}}{\sqrt{ax}}$	1;	7;	0,25	0.5; 0.75; 1.0, 1.25
14.	$y = (\sqrt{x + ae^{-x^2}})/(ax^2)$	1;	7;	0,25	0.5; 0.75; 1.0, 1.25
15.	$y = \frac{\operatorname{arctg}(\frac{x^2}{2a})}{x^3 + a}$	0;	2.5;	0,1	0.5; 1.0; 1.5, 2.0
16.	$y = \frac{\operatorname{arctg}(\frac{x}{2a})}{x^2 + 2a}$	0;	3;	0,1	0.5; 0.75; 1., 1.25
17.	$y = \frac{ax}{a + \sqrt[3]{1 + x^2}}$	-3;	3;	0,2	0.5; 1.0; 1.5, 2.0
18.	$y = \cos^2(2ax)/(3a)$	0^0 ;	360^0 ;	6^0	1; 1.25; 1.5, 2.0
19.	$y = \sin^2(ax)/(a+2)$	0^0 ;	360^0 ;	6^0	1; 1.25; 1.5, 2.0
20.	$y = a \cos(2x)/(a+4)$	0^0 ;	360^0 ;	6^0	1; 2; 3, 4
21.	$y = a^2 e^{-x}/(2 + a^2)$	-4;	4;	0,25	1; 2; 3, 4

22.	$y = \frac{\sqrt[3]{ax^2 + e^{-x}}}{ax^2}$	1; 7; 0,25	0.5; 1.0; 1.5, 2.0
23.	$y = \frac{\sqrt{x + a^2 e^{-x^2}}}{a + x}$	1; 7; 0,25	0.5; 1.0; 1.5, 2.0
24.	$y = (x^2 + a)^{\frac{xa}{x-1}}$	0; 0.8; 0,05	0.5; 1.0; 1.5, 2.0
25.	$y = (x + 4a)^{\frac{ax}{x-1}}$	0; 0.8; 0,05	0.5; 1.0; 1.5, 2.0
26.	$y = \ln^2 \left \frac{xa}{a+x} \right $	2; 12; 0,5	0.5; 1.0; 1.5, 2.0
27.	$y = \ln \left \frac{ax}{1+ax} \right $	2; 12; 0,5	0.5; 1.0; 1.5, 2.0
28.	$y = -\ln \left \frac{x+a}{1+x^2} \right $	1; 10; 0,5	1; 2; 3, 4
29.	$y = \frac{ax^2}{\sqrt{1 + ax^2}}$	-2; 2; 0,1	1; 2; 3, 4
30.	$y = -\ln \left \frac{ax}{a+x} \right $	1; 11; 0,5	1; 2; 3, 4

	A	B	C	D	E	F	G	H	I	J	K
1	Арифметическая прогрессия										
2	0	1	2	3	4	5	6	7	8	9	10
3	Геометрическая прогрессия										
4	1	2	4	8	16	32	64	128	256	512	1024
5											
6	Месяцы		Нечетные месяцы				Четные месяцы				
7	1	Январь		1	Январь		2	Февраль			
8	2	Февраль		3	Март		4	Апрель			
9	3	Март		5	Май		6	Июнь			
10	4	Апрель		7	Июль		8	Август			
11	5	Май		9	Сентябрь		10	Октябрь			
12	6	Июнь		11	Ноябрь		12	Декабрь			
13	7	Июль									
14	8	Август									
15	9	Сентябрь									
16	10	Октябрь									
17	11	Ноябрь									
18	12	Декабрь									

Рис.3.6

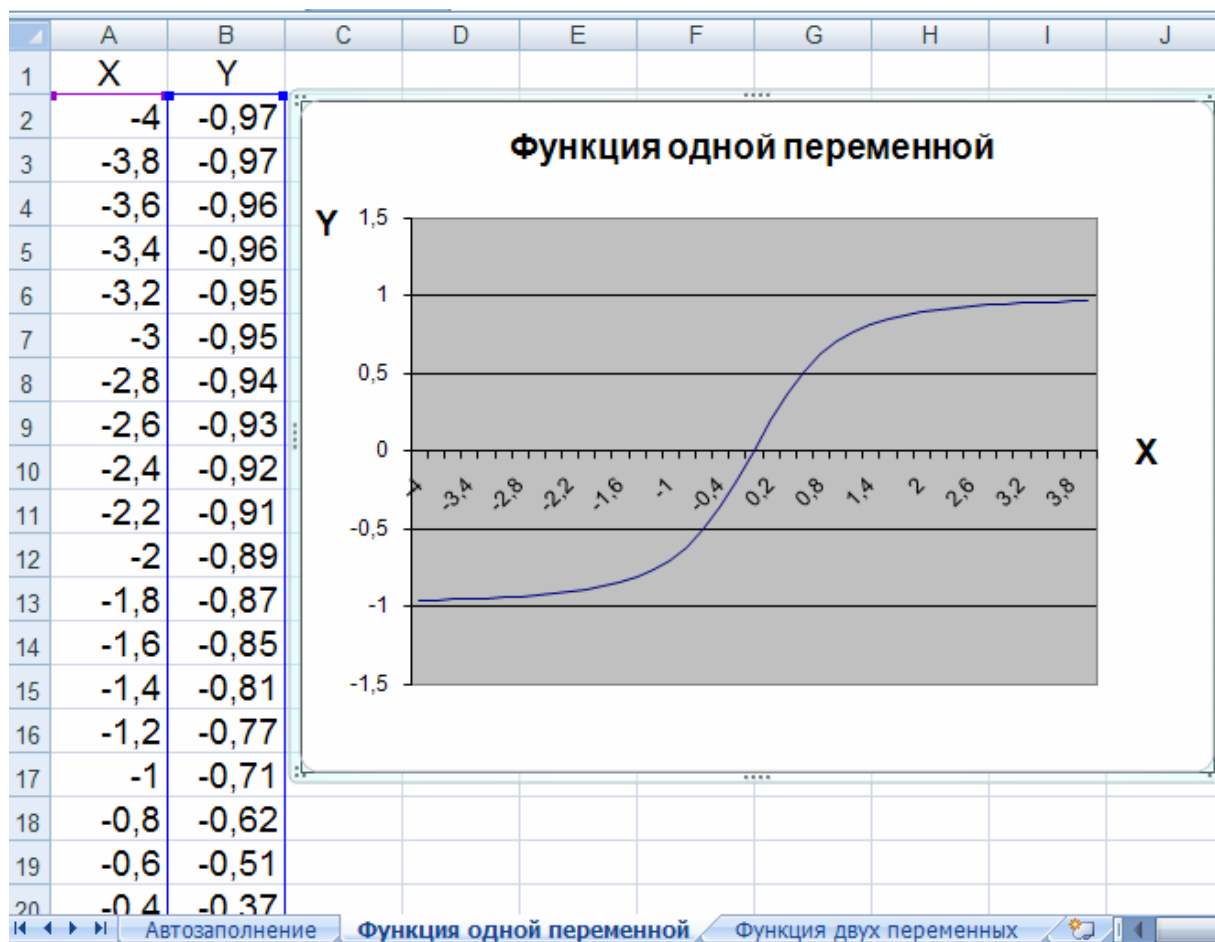


Рис. 3.7

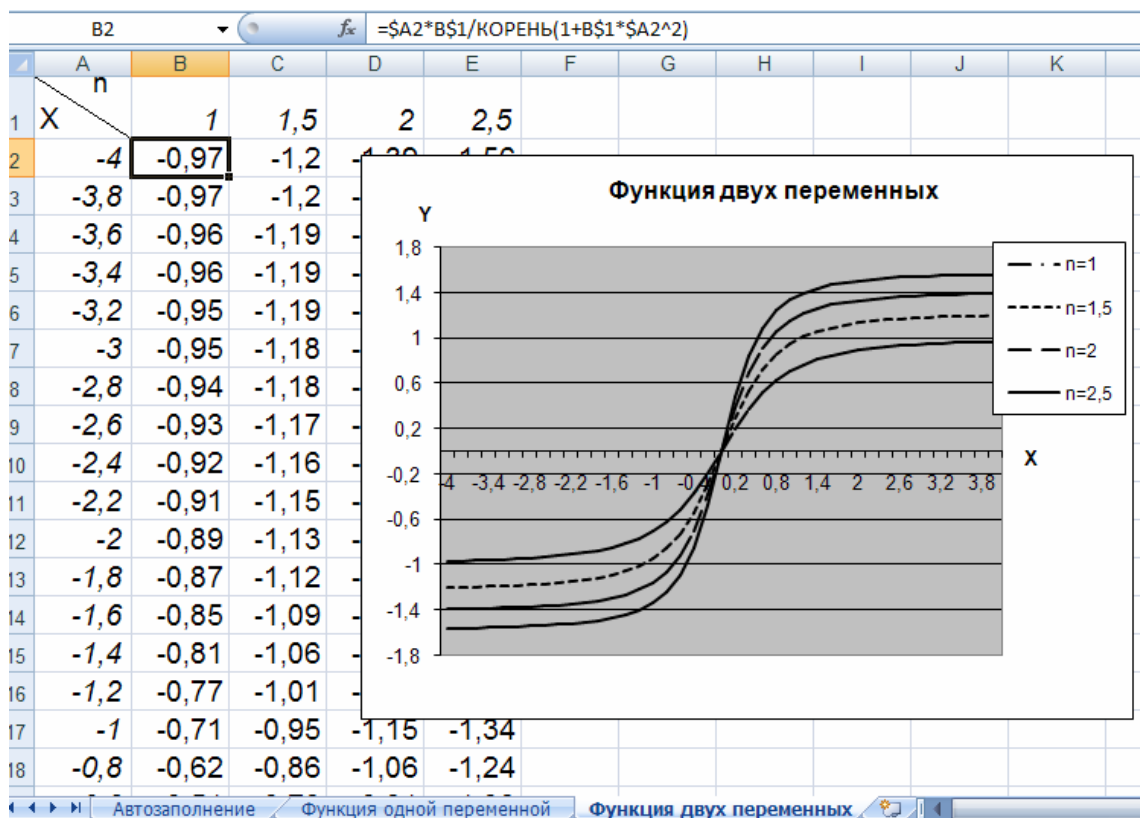


Рис. 3.8.

Лабораторная работа №4.

Система управления базой данных (СУБД) Access.

Создание таблиц и схемы данных.

1.Краткие теоретические сведения

О реляционных БД

При информационном моделировании в какой-либо предметной области рассматриваются объекты (называемые *сущностями*) и их *связи*. Состояние объектов удобно характеризовать набором существенных свойств (*атрибутов*), каждый из которых может принимать конкретные значения (из своего множества значений - *домена*). Например, объект *Товар* характеризуют атрибуты: *Наименование*, *Объём*, *Цена*. Набор значений атрибутов образует *кортеж*, например, набор (*молоко*, *1 л*, *25 р.*) – есть кортеж. Подмножество множества всех мыслимых кортежей называется *отношением* (*relation*) и обозначается (для рассматриваемого примера) как *Товар*(*Наименование*, *Объём*, *Цена*). В реляционных БД отношения представляются поименованными двумерными таблицами, в которых кортежам соответствуют строки (называемые также *записями*), а каждому атрибуту (поименованному *полю*) соответствует столбец с данными одного из типов: числового, текстового, логического и т. п. В таблице порядок следования записей несущественен, но (как и среди элементов множества) не должно быть повторяющихся записей. Больше того, в таблице обязательно должно быть поле – *простой ключ* (или несколько полей – *составной ключ*), которое однозначно определяет (идентифицирует) запись. При связывании таблиц ключевое поле 1-ой таблицы может добавляться во 2-ую таблицу, для которой оно называется *вторичным ключом*, в отличие от основного *первичного ключа* 2-ой таблицы.

Создание таблицы БД в режиме конструктора

Создание макета таблицы БД *в режиме конструктора* наиболее универсальный и хорошо контролируемый пользователем способ. Кроме этого, таблицу можно создавать *в режиме таблицы*, на основе *шаблона* и *импорта внешних данных*.

Пример 1. В СУБД Access создать макет таблицы (см. рис. 4.6), представляющей отношение *Отделы*(КодОтдела, НазвОтдела).

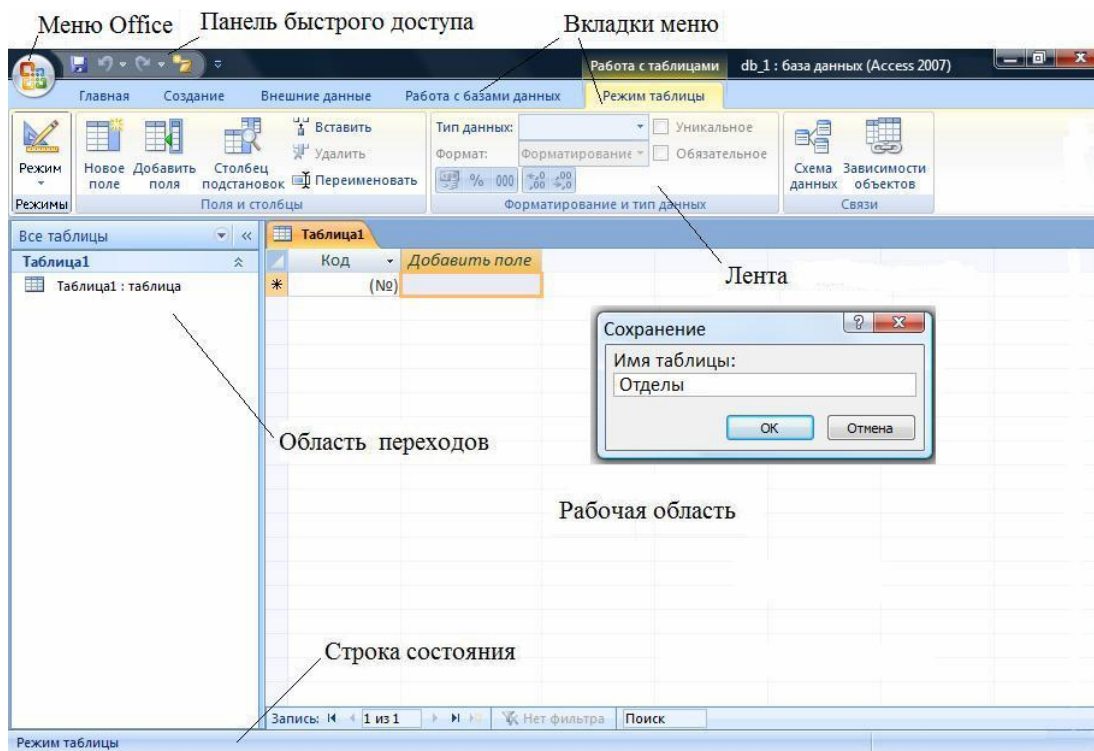


Рис. 4.1. Общий вид окна Access

При первом входе в Access следует щелкнуть (нажать и быстро отпустить левую клавишу мышки) по пиктограмме *Новая база данных*, ввести имя файла БД и щелкнуть по кнопке *Создать*. Откроется общий вид окна приложения Access с *Таблицей1* (см. рис. 4.1). Затем следует щелкнуть по кнопке *Режим*, а потом выбрать *Конструктор*. В появившемся окне *Сохранение* надо ввести имя таблицы *Отделы*, нажать ОК, и конструктор готов к созданию этой таблицы.

В 1-ую строку макета таблицы *Отделы* (см. рис. 4.2) вводят имя первичного ключа - *КодОтдела* и сразу выбирают тип данных - здесь подойдёт *числовой* (используется для всех чисел, кроме денежных сумм). В разделе *Свойства поля* указывают *Размер поля*: *целый* – отделов не много (до 8). Полезно для свойства *Условие на значение* задать, например, выражение

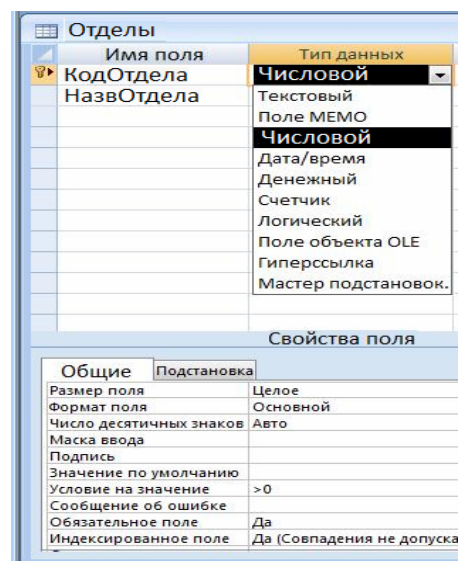


Рис. 4.2.

[КодОтдела]>0 And [КодОтдела]<9 (или >0 And <9 без указания поля), которое задаёт, что число отделов больше 0 и меньше 9. В пункте *Обязательное поле* выбирают *Да*. Полезно отметить, что поле можно *индексировать*, т.е. связать с ним вспомогательный «вектор» номеров строк, облегчающий поиск и сортировку записей.

Во 2-ую строку макета таблицы вводят имя поля: *НазвОтдела*. Выбирают тип поля – *текстовый*, а в свойствах поля указывают длину, например, 30 (по самому длинному из названий отделов), и отмечают, что это *поле обязательное* – оно не должно оставаться незаполненным. Этим завершается создание макета таблицы *Отделы*. Закрывая конструктор, следует подтвердить сохранение таблицы.

Кроме типов, выбранных выше, используют такие *типы данных*:

Денежный – он точен, его применяют для финансовых расчётов.

Логический – имеет значения «Истина/Ложь», «Да/Нет», «Вкл/Выкл».

Дата/Время – значения дат и/или времени; допускают вычисления.

Объект OLE – для подключения объектов других приложений.

Гиперссылка – для ссылки на Web-страницу, файл, адрес E-mail.

Поле Метод – для хранения текста большого объёма, комментария.

Счётчик – имеет натуральное значение, автоматически увеличивающееся, когда вставляется новая запись; используется как ключ.

Раздел конструктора *Свойства поля* содержит, кроме упомянутых, и другие средства, позволяющие не допустить в таблицы БД неверные данные. К таким средствам относятся: проверки на совпадение, обязательные поля, значения по умолчанию, маски ввода, подстановки (*мастер подстановок* использован в решении примера 2).

Пример 2. В СУБД Access создать таблицу (отношение) *Сотрудники*(КодСотрудника, ФамилияИО, КодОтдела), связанную с таблицей *Отделы* из примера 1.

Связи таблиц БД

Особенность примера 2 в том, что каждый из нескольких сотрудников *связан* со своим отделом. Таковую *связь* называют *один ко многим*

(и обозначают $1:\infty$). Для её реализации в БД *КодОтдела* указывается в каждой записи таблицы *Сотрудники*, т.е. для каждого сотрудника.

Также существует **связь один к одному** (она обозначается $1:1$), когда каждому значению атрибута 1-ой таблицы соответствует одно значение атрибута 2-ой таблицы. Например, можно выделять начальников отделов в отдельную таблицу. Тогда отдел и начальник отдела являются пример такой связи. Эту связь не обязательно реализовывать специально, достаточно в таблице отделов добавить новый столбец *НачальникОтдела*.

Встречается **связь таблиц многие ко многим** (обозначается $\infty:\infty$), например, именно так связаны Товары и Поставщики. Каждый товар могут поставлять различные поставщики, и наоборот. Эта связь моделируется в реляционных БД введением дополнительной таблицы *ТоварыПоставщиков*. В ней указывают ключ, составленный из двух полей *КодТовара* и *КодПоставщика*, а каждая из таблиц *Товары* и *Поставщики* реализует с дополнительной таблицей связь один ко многим. В Access 2007 связь **многие ко многим** можно организовать с использованием *множественного поля*. Там рассмотренный механизм связи через дополнительную таблицу скрыт от пользователя.

Решение примера 2. Итак, создавая макет таблицы *Сотрудники* примера 2, действуют как и раньше, но после ввода имени *КодОтдела* поля вторичного ключа в *типе данных* целесообразно выбрать *Мастер подстановок*. В нём надо указать таблицу *Отделы* и перенести из неё поле *КодОтдела* (нажав световую кнопку со значком $>$). При заполнении данными таблицы *Сотрудники* значения поля *КодОтдела* будут предлагаться для заполнения соответствующего столбца.

О проектировании и нормализации БД. Схема данных.

Для проектирования БД необходимо чётко представлять (а лучше и иметь описание) выбранной предметной области. Перед использованием СУБД пользователь должен определить то, какие таблицы будут в его БД, какие данные в них хранить, как связывать таблицы.

При проектировании БД стремятся избежать дублирования данных, устранить нежелательные зависимости между атрибутами и анома-

лии, которые могут приводить к нарушениям целостности данных – потерям данных и противоречиям в них при модификации данных в отдельных таблицах. Нормализация таблиц помогает устранить определённые виды аномалий модификации. Выделяют шесть нормальных форм, которым должна удовлетворять любая таблица БД.

Первая нормальная форма (1НФ) требует, чтобы таблица удовлетворяла определению *отношения* (см. выше) и, в частности, чтобы данные каждого столбца соответствовали только одному из атрибутов отношения, а в каждой ячейке находилось единственное значение. Если это так, то говорят, что таблица находится в 1НФ.

Вторая нормальная форма (2НФ) требует, чтобы не ключевые столбцы таблицы (находящейся в 1НФ) однозначно определялись всеми компонентами составного ключа, а не некоторыми из них.

Третья нормальная форма (3НФ) требует, чтобы таблица (находящаяся в 2НФ) не содержала транзитивно зависящих атрибутов (т.е. зависящих 1-ый от 2-го, 2-ой от 3-го и т.д.).

Нормальная форма Бойса-Кодда требуют наличия в таблице (находящейся в 3НФ) только одного потенциального ключа.

Четвёртая и пятая нормальные формы связаны с преобразованиями таблиц для устранения в них многозначных зависимостей.

Эти требования необходимо учитывать при разработке БД.

Задача для разработки БД

Для служебных целей сотрудникам отделов фирмы выданы мобильные телефоны (некоторым - более одного телефона). Периодически поступает таблица («распечатка») с указанием полных данных о продолжительности, времени и стоимости каждого вызова с номеров служебных телефонов сотрудников. Требуется разработать БД «ТфСвязь», по запросам к которой можно анализировать затраты на связь и то, как сотрудники используют эту телефонную связь.

Решение задачи требует использовать таблицы, рассмотренные в примерах 1 и 2. Эти таблицы представляют отношения:

Отделы(КодОтдела, НазвОтдела),

Сотрудники(*КодСотрудника, ФамилияИО, Должность, КодОтдела*).

Третья таблица – это таблица-«распечатка». Она представляет следующее отношение: **ТфРазговоры**(*НомВызова, НомТелефона,*

ДлитТфР, СтоимТфР, ДатаВремяВызова),

где *ДлитТфР, СтоимТфР* - соответственно длительность и стоимость телефонного разговора.

Необходима также таблица номеров телефонов – они закреплены за сотрудниками. Таблица представляет отношение:

Телефоны(*НомТелефона, КодСотрудника, ДатаЗакрепления*).

Важно отметить, что при увольнении сотрудника *КодСотрудника* следует очистить - заменить «пустым» значением null, которое приписывается резервным номерам телефонов. Если его номер дадут новому сотруднику, то, чтобы судить о времени пользования номером новым сотрудником, полезно добавить поле *ДатаЗакрепления*.

Между таблицами *Отделы* и *Сотрудники* есть связь один ко многим по полю *КодОтдела*. Таблицы *Сотрудники* и *Телефоны* также имеют связь один ко многим по полю *КодСотрудника*. И такая же связь по полю *КодТелефона* существует между таблицами *Телефоны* и *ТфРазговоры*. Макеты этих таблиц и связи составляют **схему данных** БД. Таблицы удовлетворяют приведённым нормальным формам, и можно в Access, создав макеты всех таблиц, построить схему данных и заполнить эти таблицы данными.

Построение схемы данных

После создания макетов всех таблиц БД могут оказаться намеченными связи, использованные в мастере подстановок. Но они требуют уточнения, а возможно потребуется установить и другие связи между таблицами. Для этого на вкладке *Работа с базами данных* в группе *Показать или скрыть* надо щелкнуть по кнопке *Схема данных*. Затем необходимо поместить в окно схемы данных все связываемые

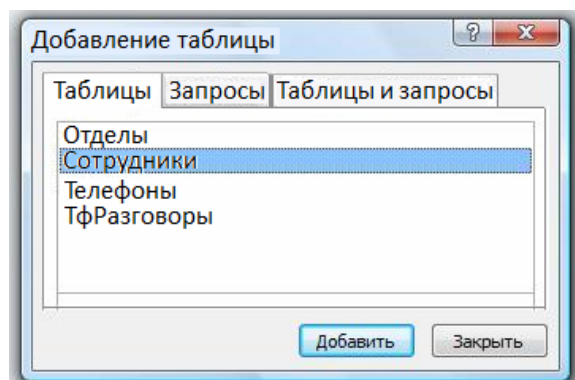


Рис. 4.3.

таблицы. Для этого в окне *Добавление таблицы* (рис. 4.3) надо указать таблицу или, нажав Ctrl, несколько таблиц, а затем нажать кнопку *Добавить*. Завершив добавление, следует *Заккрыть* окно *Добавление таблиц*. Для получения большей наглядности схемы данных расположение и размеры макетов таблиц в окне *Схема данных* можно настроить как у обычных окон.

Если связь между полями таблиц намечена (есть соединительная линия без маркировки 1:1 или 1:∞), то её следует уточнить, *кликнув* (дважды щелкнув) по соединительной линии. В открывшемся окне *Изменение связей* (рис. 4.4) следует уточнить связываемые поля, под-

твердить требование *Обеспечения целостности данных*, а также выбрать *Каскадное обновление* и *Каскадное удаление связанных полей*. Опция *Каскадное обновление (удаление)* означает, что при изменении (удалении) ключевого поля в главной таблице оно автоматически будет изменяться (удаляться) во всех подчинённых таблицах. Это исключает появление в

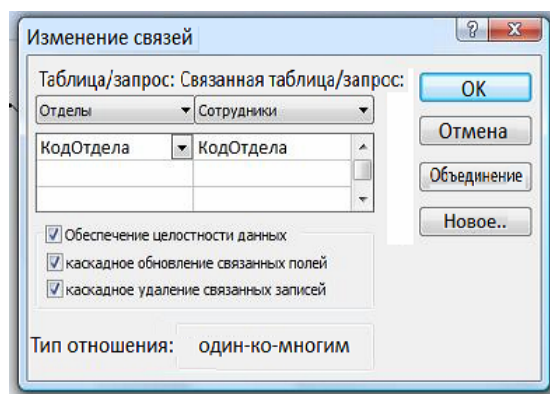


Рис. 4.4.

поле не одинакового значения. После щелчка по кнопке ОК окно закрывается и на линии связи появляется маркировка 1:1 или 1:∞.

На рис. 4.5 показана схема данных для разбираемой задачи.

Для установления (не намеченной ранее) связи таблиц надо навести курсор на одно из связываемых полей, нажать левую клавишу мышки и, не отпуская её, перевести курсор на второе связываемое поле, а за-

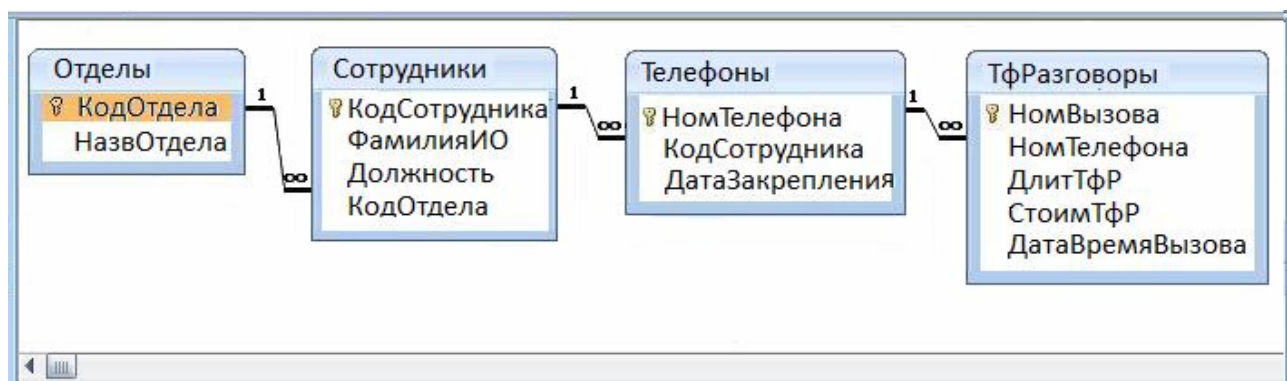


Рис. 4.5.Схема данных

тем отпустить клавишу. В открывшемся окне *Изменение связей* следует произвести описанные выше действия.

Связи таблиц можно удалять. Для этого надо щелкнуть по линии связи, а затем нажать клавишу Del и подтвердить удаление связи.

Заполнение таблиц данными

Заполнение таблицы данными производится *в режиме таблицы* построчно. На рис. 4.6 показано заполнение данными таблицы *Отделы*. Последняя строка таблицы на рис. 4.6. отмечена звёздочкой *. Эта строка предназначена для ввода новой записи – её ещё нет в таблице.

Отделы		
<input type="checkbox"/>	КодОтдела	НазвОтдела
	1	Бухгалтерия
	2	Плановый отдел
	3	Производственный отдел
	4	Отдел сбыта
*		

Рис. 4.6.

При заполнении следующей таблицы *Сотрудники* принимается во внимание её связь с таблицей *Отделы* (по полю *КодОтдела*). Для ввода значения поля *КодОтдела* будет предложено выбрать номер из списка подстановок – списка с ранее введёнными номерами отделов. Этим не следует пренебрегать, так как СУБД всё равно не позволит ввести в таблицу номер, которого нет в списке. На рис. 4.7 и 4.8 показаны заполненные данными остальные таблицы БД «ТфСвязь».

Сотрудники				
<input type="checkbox"/>	КодСотрудника	ФамилияИО	Должность	КодОтдела
	1	Рублёв Р.Р.	ст. бухгалтер	1
	2	Гарин Г.Г.	инженер	3
	3	Грошев Г.Г.	бухгалтер	1
	4	Дельцов Д.Д.	рабочий	3
	5	Платов П.П.	плановик	2
	6	Купцов К.К.	менеджер	4
	7	Пластов П.П.	ст. плановик	2
	8	Барышкин Б.Б.	менеджер	4
*				

Телефоны			
<input type="checkbox"/>	НомТелефона	КодСотрудника	ДатаЗакрепления
	8-1234-56-78	1	25.05.2010
	8-1234-56-79	1	25.05.2010
	8-1234-56-80	2	25.05.2010
	8-1234-56-81	3	25.05.2010
	8-1234-56-82	4	25.05.2010
	8-1234-56-83	5	25.05.2010
	8-1234-56-84	6	25.05.2010
	8-1234-56-85	7	25.05.2010
	8-1234-56-86	8	25.05.2010
	8-1234-56-87	8	25.05.2010
*			

Рис. 4.7.

ТфРазговоры					
<input type="checkbox"/>	НомВызова	НомТелефона	ДлитТфР	СтоимТфР	ДатаВремяВ
	1	8-1234-56-78	2,51	7,53р.	27.05.2010
	2	8-1234-56-85	3,03	9,09р.	28.05.2010
	3	8-1234-56-78	10,01	30,03р.	29.05.2010
	4	8-1234-56-83	5,5	15,15р.	29.05.2010
	5	8-1234-56-79	3,11	9,33р.	30.05.2010
	6	8-1234-56-87	7,32	21,96р.	30.05.2010
	7	8-1234-56-81	6,3	18,90р.	30.05.2010
	8	8-1234-56-80	8,24	24,72р.	30.05.2010
	9	8-1234-56-84	4,35	13,05р.	01.06.2010
	10	8-1234-56-85	5,07	15,35р.	01.06.2010
	11	8-1234-56-78	13,84	41,52р.	01.06.2010
	12	8-1234-56-87	15,67	47,01р.	01.06.2010
	13	8-1234-56-86	14,83	44,49р.	02.06.2010
	14	8-1234-56-81	3,68	11,04р.	02.06.2010
	15	8-1234-56-79	4,16	12,48р.	02.06.2010
*			0	0,00р.	

Рис. 4.8.

2.Задание к лабораторной работе

Из предложенных в варианте задания отношений создать таблицы и схему данных БД. Заполнить строки таблицы (по 8-10 строк на таблицу) заранее подготовленными данными, соответствующими предметной области.

После проверки преподавателем перенести в отчёт схему данных и по три строки из каждой таблицы.

Варианты заданий

1. **Компьютеры**(КодКомпьютера, Марка, КодПоколения, КодФирмы, Описание), **Поколения**(КодПоколения, ЭлемБаза, ТактЧастота, ВремяПоявления), **Разработчики**(КодФирмы, Название, Страна).
2. **Автомобили**(КодАвто, Марка, Класс, КодФирмы, КодСтраны, ГодВыпуска, Описание), **Фирмы**(КодФирмы, КодСтраны, Город), **Страны**(КодСтраны, Название, Столица).
3. **Методички**(НомМет, НазвМет, ГодИзд, Стр, КодАвтора), **Заказы**(КодЗаказа, СтудФИО, НомМет, КодДисц, КодАвтора), **Дисциплины**(КодДисц, НазвДисц, НомХранилища), **Авторы**(КодАвтора, ФамилияИО, Кафедра).
4. **Видеофильмы**(КодФильма, Название, КодЖанра, КодСтраны, ГодВыпуска, КодКасеты), **Жанр**(КодЖанра, Наименование),

- Страны**(КодСтраны, Название, Столица), **Кассеты**(КодКассеты, Марка, Длительность).
5. **Собаки**(КодСобаки, Кличка, Возраст, КодПороды, КодХозяина), **Собаководы**(КодХозяина, Фамилия, Имя, Отчество, ГодРождения), **ВидыПородСобак**(КодВида, НазваниеВида, Описание), **ПородыСобак**(КодПороды, Название, Рост, Вес, Заводчик, КодВида).
 6. **Книги**(КодКниги, Название, КодАвтора, КодИздательства, ЖанрКниги), **Авторы**(КодАвтора, Фамилия, Имя, Отчество, ГодРождения, ГодСмерти), **Издательства**(КодИздательства, Название, КодГорода, Примечание), **Города**(КодГорода, Название, Страна).
 7. Двигатели Внутреннего Сгорания (ДВС): **ДВС**(КодДВС, МаркаДВС, Мощность, ЧислоОбМин, КодТипаДВС, КодИзготовителя), **ТипыДВС**(КодТипа, НазвТипа, Описание), **Изготовители**(КодИзготовителя, НазвФирмы, Страна, Город).
 8. **ЗемКадастр**(КадастрНомер, АдресУчастка, Площадь, КодКатегорииУчастка, ИННВладельца), **Землевладелец**(ИННВладельца, Название, Адрес), **ПлощадиСтроений**(КадастрНомер, Дом, Сарай, Баня, ХозБлок, Прочее), **КатегорииУчастков**(КодКатегорииУчастка, НазвКатегории, СтоимЗаГа).
 9. **НалогДекларации**(КодДекларации, ИННПлательщика, ДатаПлатежа, ОблагаемаяСумма, КодНалога, КодЛьготы), **Налогоплательщик**(ИННПлательщика, Фамилия, Имя, Отчество, ГодРождения, Адрес), **Налоги**(КодНалога, НазвНалога, ПроцСтавка), **Льготы**(КодЛьготы, НазвЛьготы, ПроцСтавка).
 10. **Авиация**(КодАппарата, Марка, КодТипа, КодФирмы, КодСтраны, ГодВыпуска), **ТипыАппарата**(КодТипа, НазвТипа, Описание), **Фирмы**(КодФирмы, КодСтраны, Город), **Страны**(КодСтраны, Название, Столица).
 11. **Автопарк**(НомерАвто, Марка, Пробег, ГодВыпуска), **Водители**(КодВодителя, ФамилияИО, Класс, Стаж), **Маршруты**(НомМаршрута, Описание, Длина), **Перевозки**(Код, НомерАвто, КодВодителя, КодМаршрута, Дата).



12. **Гаражи**(НомГаража, КодТипаГаража, КодАвто, КодВладельца), **Владельцы**(КодВладельца, ФамилияИО, ГодРождения, НомАвто), **ТипыГаражей**(КодТипаГаража, Площадь, Этаж), **Авто**(НомАвто, Марка, Цвет, КодВладельца).
13. **ФутбольныеКлубы**(КодКлуба, Наименование, Город, Президент), **Тренеры**(КодТренера, Фамилия, Имя, Отчество, КодКлуба, Должность), **Игроки**(КодИгрока, Фамилия, Имя, ГодРождения, Амплуа, КодКлуба).
14. **Заказы**(КодЗаказа, ФИОЗаказчика, КодКатегории, КодТовара, КодИзгот), **Категории**(КодКатегории, НазвКатегории, НомСклада), **Товары**(КодТовара, НазвТовара, ДатаИзгот, Количество, КодИзгот), **Изготовители**(КодИзгот, НазвФирмы, Страна).
15. **Изделия**(КодИзделия, Название, КодТипоразмера, КодМатериала, ОбъёмМатериала, КодИзготовителя), **Материал**(КодМатериала, НазвМатериала, УдВес, ЦенаЗаТонну), **Изготовители**(КодИзгот, НазвФирмы, Адрес, Телефон), **Типоразмеры**(КодТипоразмера, Длина, Высота, Ширина).
16. **РежущийИнструмент**(КодИнстр, НазвИнстр, КодТипоРазм, КодМеталла, КодИзготовителя), **Типоразмеры**(КодТипоРазм, Длина, ДлинаРабЧасти, Диаметр, ДиамПосадочный, Ширина, Шаг), **Металл**(КодМеталла НазвМеталла, УдВес, КоэффПрочности), **Изготовители**(КодИзгот, НазвФирмы, Страна, Город).
17. **Договоры**(НомДоговора, НазвДоговора, ДатаНачала, ДатаОкончания, КодЗаказчика, КодОтвИсполнителя, Стоимость), **Отдел**(КодОтдела, НазвОтдела), **ОтвИсполнители**(КодОтвИсполнителя, ФамилияИО, КодОтдела), **Заказчики**(КодЗаказчика, Наименование, Адрес, Телефон).
18. Железобетонные изделия: **ЖБИ**(КодИзделия, Название, КодСостава, КодТипоразмера, КодИзготовителя), **ОбъёмНаЕдЖБИ**(КодСостава, Цемент, Песок, Гравий, Арматура), **Типоразмеры**(КодТипоразмера, Длина, Ширина, Высота, ВнешнДиаметр, ВнутрДиаметр), **Изготовители**(КодИзготовителя, Фирма, Адрес).

19. **МузПроизведения**(КодПроизведения, Название, КодАвтора, ГодСоздания, КодВидаПроизведения), **ВидыПроизведений**(КодВидаПроизведения, Название, Характеристика), **Композиторы**(КодКомпозитора, Фамилия, Имя, Отчество, Страна, ГодРождения, ГодСмерти), **ЧислоПроизведений**(КодКомпозитора, Симфоний, Опер, Концертов, Сонат, Увертюр, Этюдов, РомансовПесен).
20. **НобелевскиеЛауреаты**(КодУчёного, Фамилия, Имя, ГодРождения, ГодСмерти, КодСтраны, ГодВрученияПремии), **АвторскиеРаботы**(КодРаботы, Название, КодНауки, КодАвтора), **Науки**(КодНауки, НазваниеНауки, ВремяПоявления), **Страны**(КодСтраны, Название, Столица).

Лабораторная работа №5. Анализ информации таблиц и БД Access. Создание запросов.

1.Краткие теоретические сведения

Сортировка данных

При анализе информации удобно пользоваться отсортированными данными. Для сортировки данных таблицы необходимо выделить один или несколько соседних столбцов и на вкладке *Главная*, в группе (см. рис. 5.1) *Сортировка и фильтр* (или в контекстном меню, после нажатия правой клавиши мышки), щелкнуть по кнопке  *По возрастанию* или  *По убыванию*. После этого записи таблицы будут отсортированы сначала по крайнему левому выделенному столбцу, затем (для строк с одинаковыми значениями в левом столбце) по столбцу, расположенному правее, и так до крайнего правого выделенного столбца. Для восстановления исходного порядка следования записей в группе *Сортировка и фильтр* надо щелкнуть по кнопке *Очистить все сортировки*.

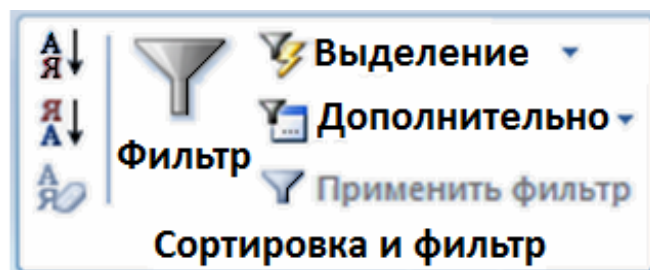


Рис. 5.1. Группа *Сортировка и фильтр* вкладки *Главная*

Фильтрация данных

При анализе информации можно установить фильтр, который позволит отображать лишь те записи, которые удовлетворяют некоторому условию фильтрации. Это условие зависит от типа и значения данных. *Обычный фильтр* позволяет выбрать одно из условий фильтрации, которые предлагает Access в ориентации на ячейку столбца, выделенную первой. То есть для его применения следует выделить ячейку столбца и нажать правую клавишу мышки, а затем (в контекстном меню) выбрать требуемое из предложенных условий. Вместо этого в группе *Сортировка и фильтр* можно щелкнуть по кнопке *Выделение* и выбирать там. Например, в подпункте *Между..* задать диапазон числовой величины или иные условия для дат. Для конкретности ниже рассмотрен пример.

Пример установки фильтра для вывода номеров телефонов, с которых выполнялись звонки 29.05.10 и 30.05.10 (суббота и воскресенье).

Решение. В таблице *ТфРазговоры* БД «ТфСвязь» выделим поле *ДатаВремяВызова*. В группе *Сортировка и фильтр* щелкнем по *Фильтру*. В меню и подменю выбираем *Фильтры дат* и *Между..*. В окне *Диапазон дат* вводим вручную (или с помощью календаря) даты *Не ранее:* 29.05.10 и *Не позднее:* 30.05.10 и щелкаем по кнопке *ОК*.

Если предлагаемые Access условия не подходят, то можно в *Расширенном фильтре* подготовить своё собственное условие. Для этого в группе *Сортировка и фильтр* надо выбрать *Дополнительно*, а в открывшемся подменю - *Расширенный фильтр*. После составления условия выполняется команда (в группе или в подменю) *Применить фильтр*. Для отмены фильтрации в секции *Сортировка и фильтр*, щелкнув по *Дополнительно*, надо выбрать *Отменить все фильтры*.

Следует отметить, что подготовка собственного условия для *Расширенного фильтра* аналогична созданию запроса в режиме конструктора, что подробно разбирается далее.

Запросы и их создание

Вообще, **запрос на выборку** – это процедура, в которой описано (в виде *условий отбора*) то, какие записи надо получить пользователю из БД. Пример условия отбора из таблицы с полем *t1* – это логическое выражение $[t1] > 0 \text{ And } [t1] < 9$ (или $> 0 \text{ And } < 9$ без явного указания поля) со значением *истина*, позволяющее получить записи, для которых $t1 \in (0; 9)$. Результатом выполнения запроса является множество записей (кортежей), удовлетворяющих условиям отбора, то есть по существу это таблицы (отношения). В Access **запрос на выборку** - это объект базы данных, который из взаимосвязанных таблиц БД (и других запросов) создаёт новую, соответствующую условиям отбора, таблицу. Правда эта таблица (отображаемая в *режиме таблицы*) временная – она существует лишь до закрытия запроса (*таблицы*).

Простые запросы на выборку можно создавать с помощью *мастера* (1-ый способ), в *режиме конструктора* (2-ой способ) предоставляется самый общий метод построения запросов, а в *режиме SQL* (3-ий

способ) можно увидеть скрытое описание запроса для Access – текст на языке *Structured Query Language* (структурированный язык запросов).

Создание запросов на выборку в режиме конструктора

Для вызова конструктора запросов на вкладке *Создание* в группе *Другие* щелкают по кнопке *Конструктор запросов*. В верхней части окна конструктора (см. рис. 5.2) отображается *часть схемы данных* со связями и таблицами, поля которых будут использоваться в запросе. В нижней части – *бланк запроса*, в котором отображаются поля ис-

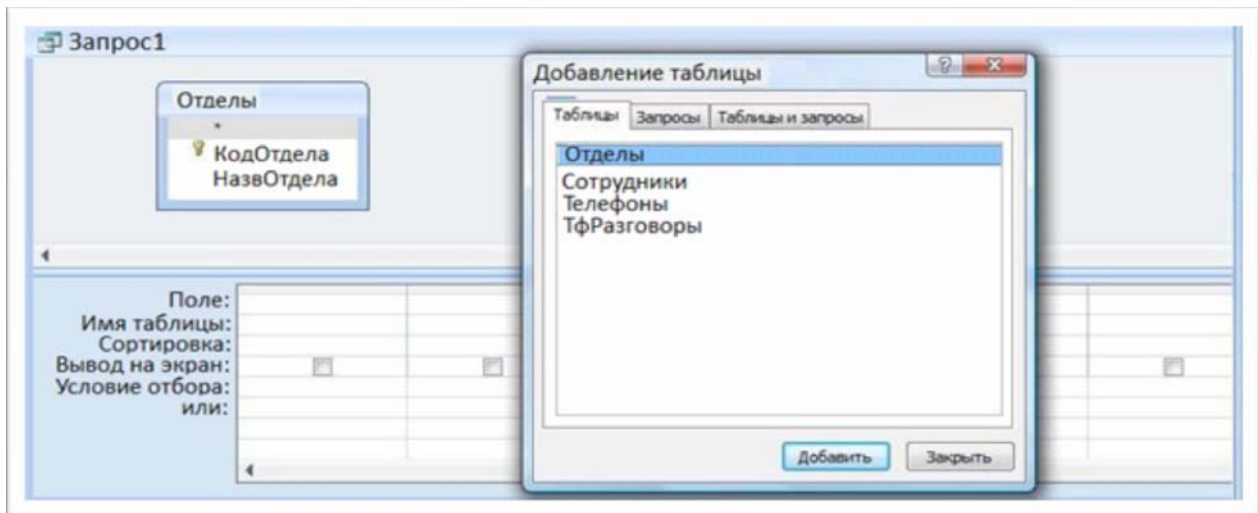


Рис. 5.2. Добавление таблиц в конструктор запросов

пользуемых таблиц и условия выборки записей. В диалоговом окне *Добавление таблицы* следует указать таблицы и запросы, которые будут использоваться в создаваемом запросе. После их выбора и щелчка по кнопке *Добавить* (или кликов по таблицам) они переносятся в верхнюю часть окна конструктора, а окно *Добавление таблицы* можно *Заккрыть*. (Для новых добавлений можно во вкладке *Создание* в группе *Настройка запроса* выбрать *Отобразить таблицу*.)

Затем необходимо выбрать поля таблиц и включить их в запрос. Это можно сделать одним из способов: кликнув по выделенному полю, перетаскиванием полей в бланк запроса или использовать раскрывающийся список в ячейках верхней строки бланка запросов. Кликнув по звёздочке можно перенести все поля таблицы в бланк запроса. Однако рекомендуется включать в запрос лишь самые необходимые поля. Не следует дважды включать одно и то же поле.

Выделенный столбец с полем можно удалить, нажав клавишу Del. Для выделения столбца курсор подводят к верху столбца до появления стрелки, указывающей вниз, и щелкают мышкой. Поля следует располагать в бланке запроса слева направо в таком порядке, в каком вы хотите их видеть на экране в результатах выполнения запроса. Для изменения порядка следования столбец можно перетащить – выделить его, а затем нажать левую клавишу мышки и, не отпуская её, отбуксировать чёрную полосу-указатель в нужное место. Имя выводимого поля можно заменить *псевдонимом*. Для этого имя-псевдоним с символом двоеточия «:» в конце надо набить перед наименованием поля в верхней строке бланка запроса.

В строке *Сортировка* можно указать поле и порядок сортировки по этому полю. Если ни для одного поля не сделано отметок, записи будут выводиться в порядке их следования в БД.

Если вам не нужно выводить на экран одно или несколько полей, то следует сбросить для них флажок *Вывод на экран*.

В строке *Условие отбора* и в строке *Или* задают логическими выражениями условия отбора записей. Логические выражения, записанные в одной строке бланка, но в разных столбцах, соединяются оператором And (И). Если нужно получить объединение строк, задаваемое применением к выражениям операторов Or (ИЛИ), то эти выражения размещают в разных строках бланка запроса.

В условиях отбора используются *выражения*. Они состояются из *операндов*, *операторов* (знаков операций), возможно, круглых скобок и задают правило вычисления этого выражения. Операндами - объектами, участвующими в операциях - могут быть константы, идентификаторы, функции и выражения. Примеры выражений:

[ТфРазговоы]![Дата] = Date() – 7

[Тариф]*[ДлитТфР] > 25,00р.

В приведённых выражениях: [ТфРазговоы]![Дата], [Тариф], [ДлитТфР] – идентификаторы, первый включает имя таблицы и имя поля, два других – только имена полей (подразумеваемой таблицы);

Date() – функция, вычисляющая текущую дату;

=, −, > операторы отношения (по краям) и вычитания (в середине);
7 и 25,00р. – константы.

При составлении выражений можно использовать *Построитель выражений*. Для этого надо на вкладке *Конструктор*, в группе *Настройка запроса*, выбрать *Построитель* или щёлкнуть правой клавишей мышки ячейку, куда требуется поместить выражение, и выбрать *Построить...* *Построитель выражений* позволяет вставлять имена таблиц и полей без их набивки с клавиатуры, находить с помощью обзора требуемые функции. *Построитель* содержит перечень операторов, который включает: арифметические, логические, операторы слияния и специальные операторы.

Арифметические операторы используются для вычисления значений от двух или более чисел или для изменения знака числа.

<i>Оператор</i>	<i>Описание</i>	<i>Пример</i>
+	Сложение	[Сумма]+25,00р.
−	Вычитание или минус	Date() □ 7
*	Умножение	[Тариф] * [Расход]
/	Деление	[Сумма] / [Количество]
\	Деление нацело	[Количество] \ 3
Mod	Остаток от деления	12 Mod 5
^	Возведение в степень	[Длина]^2

Логические операторы дают значения True (истина) или False (ложь).

<i>Оператор</i>	<i>Описание</i>	<i>Пример</i>
And	Конъюнкция (И)	a And b
Or	Дизъюнкция (ИЛИ)	a Or b
Eqv	Эквивалентность	a Eqv b
Not	Отрицание (НЕ)	Not a
Xor	Исключающее ИЛИ	a Xor b

Операторами слияния объединяют текстовые значения в единое.

<i>Оператор</i>	<i>Описание</i>	<i>Пример</i>
&	Конкатенация	"Смирнов" & "и сыновья"
+	Конкатенация	"Смирнов" + "и сыновья"

Специальные операторы используются так, как показано ниже:

Оператор	Определяет:	Пример
Is	Равен ли или не равен аргумент пустому значению Null ?	Is Null или Is Not Null
Between	Принадлежит ли числовое значение указанному отрезку [10; 20] ?	Between 10 And 20
In	Содержится ли строковое значение в списке?	In("Рим", "Париж")
Like	Содержит ли строковое значение указанные символы? ¹	Like "Видео*" Like "???db"

¹ Значение включает подстановочные знаки: ? - любой символ,
* - произвольное число любых символов.

Примеры построения запросов на выборку

Построение Запроса1 на выборку из одной таблицы ТфРазговоры БД «ТФСвязь», рассмотренной в лабораторной работе №4. Пусть требуется вывести в возрастающем порядке все номера телефонов, с которых инициировались разговоры длительностью более 10 минут, и даты этих разговоров.

Решение. При построении запроса таблицу *ТфРазговоры* переносим в верхнее окно конструктора. В столбцах 1, 2 и 3 указываем соответственно поля: *НомТелефона*, *ДлитТфР*, *ДатаВремяВызова*. Для 1-го поля в строке *Сортировка* указываем: *по возрастанию*. Для 2-го столбца в строке *Условие отбора* задаём: *>10*.

Запрос в режиме конструктора показан на рис. 5.3. Для его сохране-

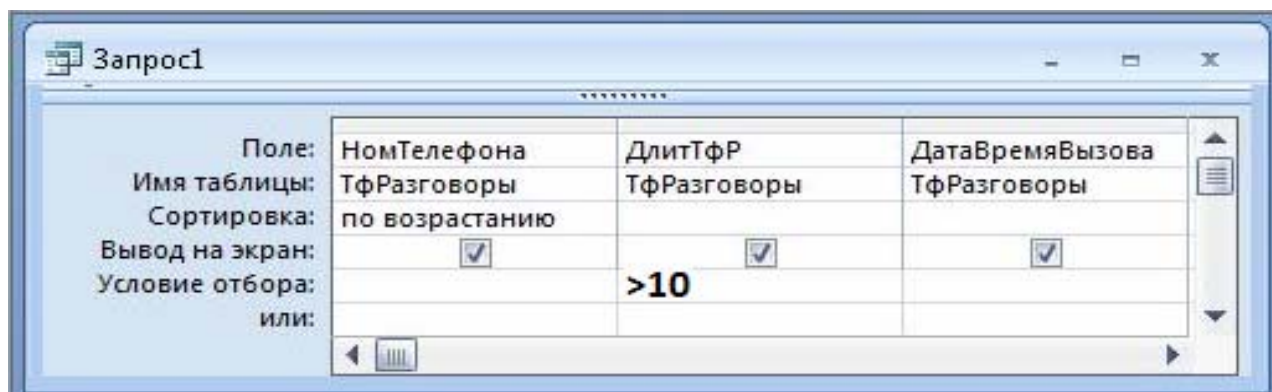
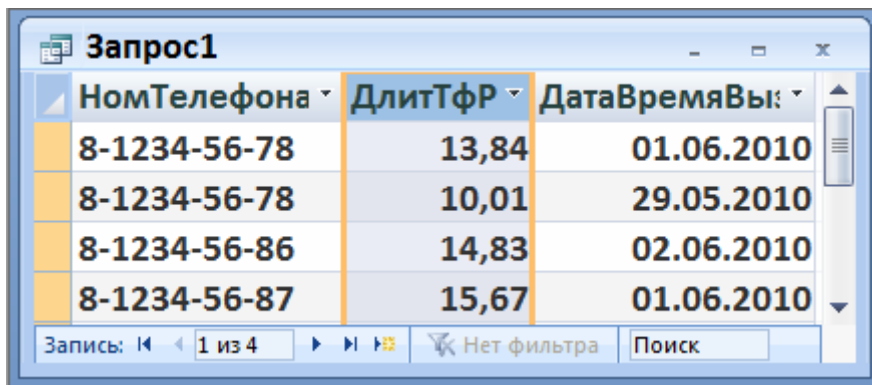


Рис. 5.3. *Запрос1* в режиме конструктора

ния в меню *Office* выбираем *Сохранить* и затем в окне *Сохранение* подтверждаем имя *Запрос1* или вводим новое имя. Запрос с указанным именем появится в области переходов. Для выполнения запроса достаточно кликнуть по нему. Результат выполнения запроса показан на рис. 5.4.



НомТелефона	ДлитТфР	ДатаВремяВык
8-1234-56-78	13,84	01.06.2010
8-1234-56-78	10,01	29.05.2010
8-1234-56-86	14,83	02.06.2010
8-1234-56-87	15,67	01.06.2010

Рис.5.4. *Запрос1* (режим таблицы)

Полезно отметить, что таким же образом устанавливается расширенный фильтр, дающий такую же выборку. Разница в том, что запрос сохраняется, и для изменённой таблицы его можно перезапустить, а фильтр приходится устанавливать заново.

Запрос можно корректировать. Для этого следует выбрать запрос в области переходов и, нажав затем правую кнопку мышки, щелкнуть в контекстном меню по позиции *Конструктор*. Или при выполненном запросе на вкладке *Главная* в группе *Режимы* выбрать *Конструктор*.

Режим SQL можно выбрать там же, в группе *Режимы*. Для нашего запроса оператор языка SQL имеют следующий вид:

```
SELECT ТфРазговоры.НомТелефона, ТфРазговоры. ДлитТфР
FROM ТфРазговоры
WHERE (((ТфРазговоры. ДлитТфР)>10)),
FROM ТфРазговоры
ORDER BY ТфРазговоры.НомТелефона;
```

SQL декларативный язык – он описывает то, что требуется отобразить из БД, но он не задаёт последовательность вызовов процедур, которые должны выполнять этот отбор. Оператор SELECT (Выбрать) указывает идентификаторы полей, по которым будет производиться отбор, из (FROM) таблицы. А после служебного слова WHERE (где) записывается условие отбора – фильтр записей. И наконец, после ORDER BY (упорядочить по) указано поле, по которому выполняется сортировка. Вообще говоря, запросы можно писать и сразу на SQL в окне редактора. Но чаще используют конструктор, а при необходимо-

сти составленное им корректируют в режиме SQL. Например, если использовать все столбцы таблицы, то можно 1-ую строку сократить до `SELECT *`. При выборке из нескольких таблиц их имена перечисляются через запятую после `FROM`.

Построение Запроса2 на выборку из БД «ТФСвязь», рассмотренной в лабораторной работе №4. Пусть требуется определить фамилии и отделы всех сотрудников, у которых за последнюю неделю длительность хотя бы одного телефонного разговора (с вызовом) превышала 10 минут. Фамилии вывести в алфавитном порядке.

Решение. Для построения запроса используем все таблицы: *Отделы*, *Сотрудники*, *ТфРазговоры*, *Телефоны*. В столбцах указываем поля: *ФамилияИО*, *ДлитТфР*, *НазвОтдела*, *ДатаВремяВызова*, а для наглядности вывода к ним слева (без кавычек) приписываем псевдонимы: «Фамилия_И_О:», «Время (мин):», «Отдел:», «Дата:».

В 1-ом столбце (с полем *ФамилияИО*) в строке *Сортировка* указываем *по возрастанию*. Во 2-ом столбце (с полем *ДлитТфР*) записываем условие: `>10`. Когда сравнивается значение поля столбца, в котором записано условие, то явно указывать идентификатор не надо, и нет необходимости писать: `[ДлитТфР] > 10`. В 4-ом столбце записываем условие: `Date() - 7`, по которому отбираются только записи с датами

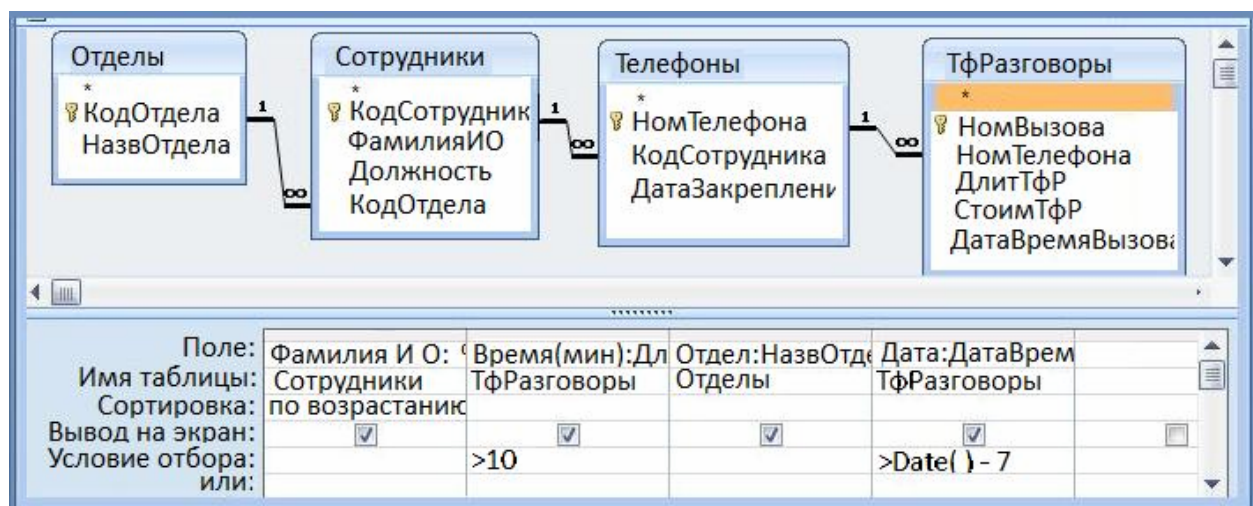


Рис. 5.5. Запрос2 в режиме конструктора

вызова, превышающими текущую дату минус семь дней (неделю). Для выбираемых строк эти условия должны выполняться одновременно, то есть рассматривается пересечение множеств записей, отби-

раемых по каждому из условий. Поэтому условия размещаются в одной строке, то есть используется оператор And (И).

Запрос в режиме конструктора (представленный на рис. 5.5) сохраняем в БД под именем *Запрос2*.

В режиме SQL Запрос2 имеет следующий вид:

```
SELECT Сотрудники.ФамилияИО AS [Фамилия_И_О], ТфРазговоры.ДлитТфР
AS [Время(мин)], Отделы.НазвОтдела AS Отдел,
ТфРазговоры.ДатаВремяВызова AS Дата
FROM ((Отделы INNER JOIN Сотрудники ON Отделы.КодОтдела =
Сотрудники.КодОтдела) INNER JOIN Телефоны ON
Сотрудники.КодСотрудника = Телефоны.КодСотрудника) INNER JOIN
ТфРазговоры ON Телефоны.НомТелефона = ТфРазговоры.НомТелефона
WHERE (((ТфРазговоры. ДлитТфР)>10) AND ((ТфРазговоры.
ДатаВремяВызова) > Date()-7))
ORDER BY Сотрудники.ФамилияИО;
```

Можно заметить, что при указании полей добавились альтернативные имена после слова AS (как). А после FROM для таблиц описаны внутренние связи (INNER JOIN) по (ON) указанным полям.

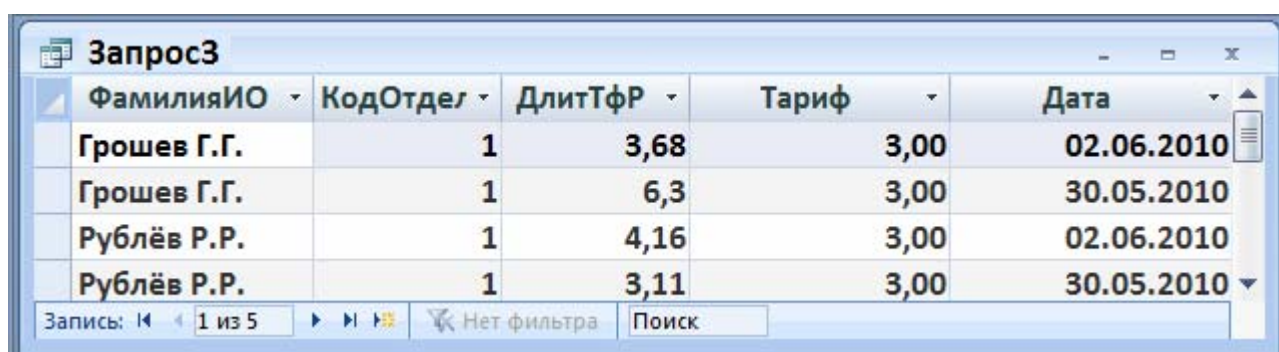
Построение Запроса3 с вычисляемым полем.

Пусть требуется из БД «ТфСвязь» по звонкам сотрудников 1-го отдела с 30.05.10 по 02.06.10 сверить тарифы, то есть получить стоимость одной минуты телефонного разговора.

Решение. Здесь также как в предыдущем запросе используем все таблицы. Понятно, что в столбце с полем *КодОтдела* надо записать *Условие отбора* отдела: =1. А для столбца с полем *ДатаВремяВызова* в строке *Условие отбора*: Between #30.05.10# And #02.06.10# (или >#29.05.10# And <#03.06.10#). Особенностью этого запроса является то, что здесь необходимо получить новое поле с результатами вычисления – деления стоимости звонка на время разговора. Для этого в

Рис. 5.6. Запрос3 в режиме конструктора

поле пустого столбца следует набить выражение: $[СтоимТфР]/[ДлитТфР]$, а вместо выведенного слева слова *Выражение1*: записать название поля *Тариф*: (рис. 5.6). Полезно назначить формат вывода в это поле. Для этого надо встать в ячейку этого столбца, на вкладке *Создание* в группе *Показать или скрыть* щелкнуть по кнопке *Страница свойств*, на вкладке *Общие* *Окна свойств* выбрать *Формат поля: фиксированный* и *Число десятичных знаков: 2*, а потом закрыть окно свойств и закрыть (с сохранением запроса) конструктор. При выполнении этого запроса в выводимой таблице



ФамилияИО	КодОтдел	ДлитТфР	Тариф	Дата
Грошев Г.Г.	1	3,68	3,00	02.06.2010
Грошев Г.Г.	1	6,3	3,00	30.05.2010
Рублёв Р.Р.	1	4,16	3,00	02.06.2010
Рублёв Р.Р.	1	3,11	3,00	30.05.2010

Рис. 5.7. *Запрос3 (режим таблицы)*

появится новое поле *Тариф* (рис. 5.7). Важно отметить, что в исходных таблицах никаких новых полей заведено не будет.

Построение *Запроса4* с параметром. Пусть требуется из БД «ТфСвязь» вывести сведения по телефонным звонкам какого-либо сотрудника. Фамилия_И_О сотрудника должна указываться в процессе выполнения запроса.

Решение. Для выполнения требований задачи в столбец с полем *ФамилияИО*, в строку *Условие отбора*, необходимо записать параметр – заключённый в квадратные скобки текст, который будет использоваться как подсказа при вводе. В данном случае уместно записать $[Фамилия_И_О:]$, а после текста поставить двоеточие, точки внутри имени не допустимы. При выполнении такого запроса в появившееся окно *Введите значение параметра* будет выведена наша подсказка *Фамилия_И_О:*. В окне ввода (см. рис. 5.9) следует задать фамилию и инициалы конкретного сотрудника в том виде, как они записаны в БД, и щелкнуть по кнопке *ОК*. Введённое в запрос значение будет использовано для отбора записей.

Итоговые запросы (запросы с групповыми операциями)

Для вычисления итоговых значений по группам данных используются запросы с групповыми операциями. Строка *Групповая операция* в бланке запроса позволяет выделить группы записей с одинаковыми значениями в указанных полях и вычислить итоги для этих групп по одной из статистических функций.

Построение Запроса5 с групповыми операциями. Пусть требуется из БД «ТфСвязь» вывести для какого-либо сотрудника итоговые сведения: общий счёт и общее время по его телефонным разговорам, а также среднюю продолжительность одного, инициированного им, телефонного разговора. Фамилия_И_О сотрудника должна указываться в процессе выполнения запроса.

Решение. Фамилия сотрудника может фигурировать неоднократно в результирующей таблице предыдущего *Запроса4*. Поэтому возможно произвести группировку по полю *ФамилияИО* и выполнить требуемые подсчёты в соответствующих полях сгруппированных записей. Сначала, как всегда, помещаем все связанные таблицы в верхнюю часть окна *Конструктора*. Затем в ленте, в группе кнопок *Показать или скрыть*, щелкаем по кнопке *Итоги* (со значком Σ), и в бланке запроса появляется новая строка *Групповая операция*: . В 1-ый столбец помещаем поле *ФамилияИО* – столбец заполняется как в предыдущем запросе, но в строке *Групповая операция*: должна стоять *Группировка*. В первую строку 2-го столбца помещаем (вместе с псевдонимом) поле *СтоимТфР*, то есть *ОбщСумма: СтоимТфР*, а в строке *Групповая операция* выбираем функцию Sum для суммирования значений стоимостей разговоров группы. Таким же образом заполняем 3-ий столбец с псевдонимом и полем *ОбщееВремя(мин): ДлитТфР*, в котором функция Sum вычисляет общее время по всем записям группы. И следующий 4-ый столбец заполняем именами *СрВремя(мин): ДлитТфР*, но в строке *Групповая операция* указываем функцию Avg, которая вычисляет среднее арифметическое время одного разговора. Для полей столбцов со 2-го по 4-ый целесообразно установить формат вывода. Для иллюстрации вычислений с функциями заполним первую строку 5-го столбца так:

СрВремя: Sum([ДлитТфР])/Count([ДлитТфР]) ,

а в строке *Групповая операция* укажем *Выражение*. Для удобства набивки в поле длинных выражений (дав Shift+F2) можно использовать окно *Область ввода* или *Построитель выражений*. В приведённом выражении среднее время определяется как вычисляемое поле с ис-

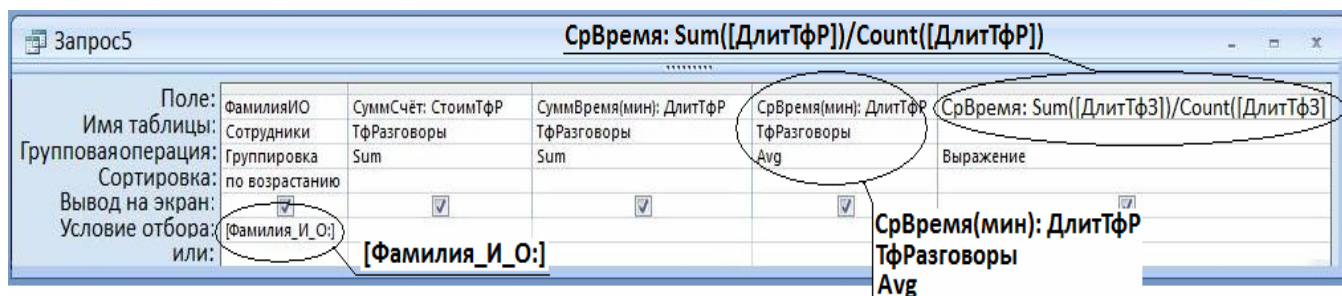


Рис. 5.8. *Запрос5* в режиме конструктора

пользованием функций Sum и Count (количество) от поля *ДлитТфР*.

Вид *Запроса5* в *Конструкторе* дан на рис. 5.8; ввод значения параметра показан на рис. 5.9, а результат выполнения представлен на рис. 5.10.

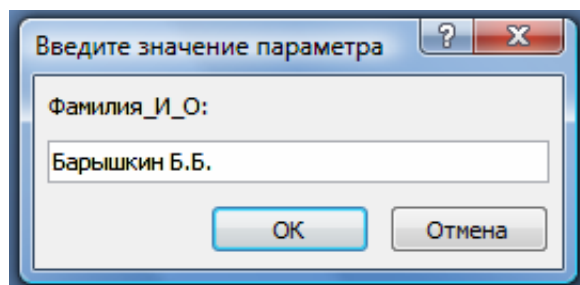


Рис. 5.9. Ввод значения параметра

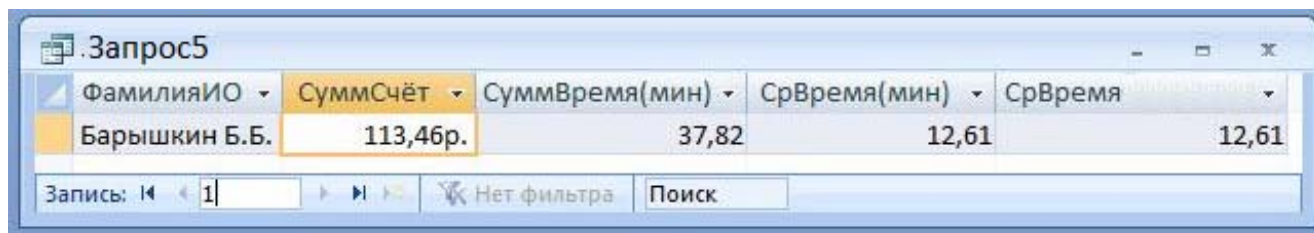


Рис. 5.10. *Запрос5* (режим таблицы)

2.Задание к лабораторной работе

Для получения информации из БД, созданной в предыдущей лабораторной работе, подготовить, согласовать с преподавателем и затем построить в конструкторе три запроса:

- запрос на выборку с критериями поиска;
- итоговый запрос или запрос, формирующий вычисляемое поле;
- параметрический запрос.

Предъявить преподавателю результаты выполнения этих запросов. Бланки запросов (режим конструктора) изобразить в отчёте. Один из запросов рассмотреть в режиме SQL, списать текст запроса в отчёт. В отчёте пояснить смысл элементов SQL-запроса.

Лабораторная работа №6.

Начала алгоритмизации. Обработка в Excel табличных данных с использованием макросов на VBA.

1.Краткие теоретические сведения

Макросы и автоматизация обработки данных

При работе, например, в Проводнике, в Word или Excel для выполнения какого-либо действия нажимают соответствующую кнопку, инициирующую это действие. Однако при частом нажатии одинаковых последовательностей кнопок рутинная работа утомляет и отнимает неоправданно много времени. В таких случаях целесообразно уяснить последовательности повторяющихся действий (*алгоритм*) и записать *программу* – последовательность команды для выполнения этих действий компьютером. Программу можно оформить в виде так называемого *макроса*, «запомнив» производимые нажатия клавиш в режиме записи макроса или составив макрос самостоятельно из команд внутреннего языка приложений. *Выполнение* программы (макроса) позволит получить результат в автоматическом режиме. Встречаются действия, для которых кнопка вообще не предусмотрена. В этих ситуациях также выручает *алгоритмический подход* – составление алгоритма и соответствующей программы (макроса) для выполнения требуемого действия. Для запуска макроса можно создать новую кнопку или связать запуск с нажатием комбинации клавиш.

Основы алгоритмизации с использованием макросов

Последовательность действий для решения, например, вычислительной задачи, то есть алгоритм, можно записать обычными словами с использованием математической символики. Часто алгоритмы представляют в виде блок-схем (наглядного графического изображения). Важно отметить, что такие записи непосредственно не воспринимаются компьютером, однако является основой для написания программы, которую может выполнить компьютер. Далее рассматриваются на примерах важнейшие типы алгоритмов.

Основные типы вычислительных алгоритмов

Линейные вычислительные процессы

Пример 1. Заданы высота H и радиус R основания прямого кругового конуса. Вычислить объём V конуса.

Основу алгоритма вычисления составляет формула $V = \frac{H}{3} \cdot 2\pi R^2$, а все действия (алгоритм), связанные с расчётом, можно представить так:

Начало алгоритма

Назначить конкретные значения H и R

Вычислить объём по формуле $V = \frac{H}{3} \cdot 2\pi R^2$

Отобразить результаты (значение V)

Конец алгоритма

В этом алгоритме действия выполняются последовательно друг за другом, такие алгоритмы называются *линейными*.

Этот алгоритм можно записать в виде макроса Excel на VBA, а значения H и R поместить, например, в ячейки A1 и B1 или просто задать в теле макроса.

Для создания макроса выбирают на вкладке *Вид* группу *Макросы*, задают имя, например V_k , и выбирают *Создать*. Появляется окно редактора VBA (Visual Basic for Application – визуального Бейсика для приложений), показанное на рис. 6.1. В окне обозревателя проектов виден включенный в проект модуль Module1 – в нём создаётся макрос. В окне кода уже записаны строки начала Sub $V_k()$ и конца End Sub *процедуры макроса* (алгоритма), которые надо дополнить остальными строками алгоритма так, чтобы получился текст макроса:

Sub $V_k()$

$H = 4.5 : R = 3.0$ ' Назначить конкретные значения H и R . Строка 2.

$V = H/3 * 3.14 * R^2$ ' Вычислить объём конуса. Строка 3.

Debug.Print V ' Отобразить результаты (значение V). Строка 4.

End Sub

В строке 2 текста макроса переменным Н и Р назначаються (присваиваются) в двух операторах конкретные числовые значения. Её можно записать, указав явно имена ***операторов присваивания*** **Let** (пусть):

Let Н = 4.5 : **Let** Р = 3.0 ' Назначить конкретные значения Н и Р. Строка 2, которые обычно в Бейсике по аналогии с другими языками программирования не указываются. По правилам Бейсика операторы, записанные в одной строке (как здесь во 2-ой строке), отделяются друг от друга двоеточием. Справа после апострофа «'» можно дать примечания, которые выводятся на экран зелёным цветом, а в нашем тексте они будут выделяться курсивом. Ключевые слова (названия операторов) изображаются на экране синим цветом, а в наших текстах они выделяются полужирным шрифтом.

В строке 3 в операторе присваивания сначала вычисляется значение, стоящее справа от равенства, а затем это значение приписывается (присваивается) переменной V, стоящей слева от равенства. В строке 4 осуществляется вывод вычисленного значения V объёма методом Print (печать) объекта Debug (отладчика) в окно Immediate Window (окно отладки или ближе к оригиналу - окно «срочно»). Оно вызывается из пункта меню View (или по Ctrl+G). Непосредственные вычисления можно выполнять в окне отладки, например, вычислить предполагаемый результат работы макроса. Для

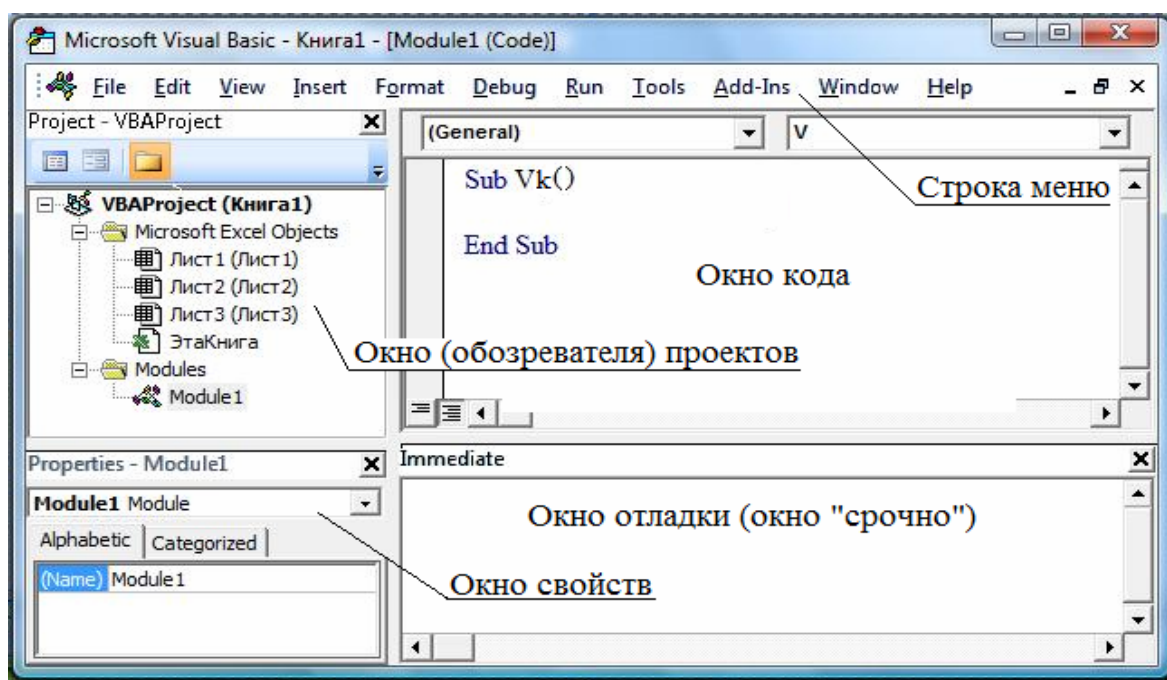



Рис. 6.1.Окно редактора VBA

чего в этом окне следует набить: $? (4.5/3)*3.14 *(3.0^2)$ и дать «Enter». Знак «?» вопроса перед выражением означает печать (Print), и по этому указанию на следующей строке будет выведен результат 84,78. Такой же результат можно получить, набив построчно операторы 2-ой и 3-ей строк макроса и выполнив затем печать вычисленного объёма по $? V$ «Enter». Полезно отметить, что тот же результат получается и по записи $? 4.5/3*3.14 *3.0^2$ без простановки скобок, т. е. в Бейсике возведение в степень «^» имеет приоритет. После возведения в степень выполняются слева направо деление и умножения как операции одного ранга при отсутствии скобок.

Выполнение макроса в автоматическом режиме можно инициировать через пункт меню Run, нажатием кнопки запуска  или клавиши F5. При первом знакомстве больше подходит выполнение в режиме отладки операторов макроса: через пункт меню Debug и/или нажатием клавиши F8. Причём курсор должен находиться внутри макроса. Тогда подсвечивается (обычно жёлтым цветом) оператор, который предстоит выполнить. Наведя указатель мышки на переменную H в первом операторе присваивания, можно увидеть её значение и отметить, что до выполнения оператора переменная имела значение «Empty» (пусто), а после выполнения (по очередному нажатию F8) значение изменилось на 4,5. После выполнения оператора 4-ой строки в окно отладки выводится результат 84,78.

Исходные значения H и R удобно задать в ячейках листа Excel, например, соответственно в ячейках A1 и B1. Для ссылки в тексте макроса на ячейки используется нотация (r, c) – строка (row), столбец (column). Например, ячейку B1 можно в такой нотации задать как (1, 2). В макросе на эту ячейку рабочего листа ссылаются, записывая Cells(1, 2). В таком варианте текст макроса получает следующий вид:

Sub Vk()

H = Cells(1,1): R = Cells(1,2) ' Выбрать значения H и R из ячеек A1, B1

V = H/3*3.14 * R^2 ' Вычислить объём конуса

Debug.Print V: Cells(1,3) = V ' Отобразить результаты (значения V)

End Sub

Результат выводится и в окно отладки и в ячейку C1 рабочего листа.

С именем макроса можно связать сочетание клавиш или световую кнопку, по которым он будет запускаться. Выбрав при входе в макрос *Параметры* и задав, например, сочетание клавиш Ctrl+Shift+V, получаем возможность удобно запускать макрос по этому сочетанию.

Задание для самоконтроля. Поместить в ячейку B1 значение R, равное 3,0. Задавая в ячейке A1 значения H и вызывая по сочетанию клавиш выполнение макроса, получить в ячейке C1 объём 100 .

Ветвящиеся вычислительные процессы

Пример 2. По заданным значениям коэффициентов p, q квадратного уравнения $x^2 + px + q = 0$ вычислить значения вещественных корней или вывести сообщение «нет корней» об их отсутствии.

Алгоритм решения задачи можно представить наглядно, в виде *блок-схемы* примера 2 с записью этапов алгоритма в геометрических фигурах, соответствующих типу этапа.

После ввода коэффициентов и вычисления дискриминанта D следует *проверка условия* (заключено в ромб), в зависимости от выполнения или не выполнения которого выбирается направление дальнейших действий. Блок-схема примера 2 в этом месте разветвляется – отсюда и название «ветвящиеся вычислительные процессы». При не отрицательном D вычисляются и выводятся значения корней, а при отрицательном выводится сообщение «нет корней».



Алгоритм реализован в виде макроса Korn1 на VBA. Значения коэффициентов p, q берутся соответственно из ячеек A1, B1, значения корней - выводятся в ячейки C1, D1, а символьная константа "Нет корней" помещается в ячейку C1.

Sub Korn1() ' Текст макроса

$p = \text{Cells}(1, 1): q = \text{Cells}(1, 2)$

$D = (p / 2) ^ 2 - q$

If D < 0 Then

Cells(1, 3) = "Нет корней" ' Ветвь «да»

Else

x1 = -p / 2 - Sqr(D) ' Ветвь «нет»

x2 = -p / 2 + Sqr(D)

Cells(1, 3) = x1: Cells(1, 4) = x2

End If

End Sub

В приведённом макросе условие $D < 0$ записано в **блочном условном операторе If** (если). Ветвь «да» блок-схемы размещается между ключевыми словами **Then** (тогда) и **Else** (иначе) этого оператора, а ветвь «нет» - между **Else** и завершающим блок **End If** (конец если). После выполнения каждого из блоков «да» и «нет» происходит выход за **End If**, здесь на завершающий макрос оператор **End Sub**.

Для вычисления арифметического корня из D использована встроенная функция с именем **Sqr**. После имени функции всегда открывается скобка, затем записывается аргумент и скобка закрывается.

Первое выполнение макроса полезно выполнить в режиме отладки. Конкретные коэффициенты p , q при этом лучше назначить непосредственно в тексте макроса, записав, например, во второй строке:

p = -5: q = 6 ' p = Cells(1, 1): q = Cells(1, 2)

Выполняя макрос по шагам (по F8), полезно проследить за изменением значений переменных и последовательностью выполнения операторов при разных исходных данных p , q . Уместно также дополнить ветви «да» и «нет» соответственно операторами

Debug.Print p, q, "Нет корней" и **Debug.Print** p, q, x1, x2

вывода в окно отладки.

При выполнении отлаженного макроса в ячейки A1, B1 помещаются значения коэффициентов, запускается макрос, и результаты просматриваются в ячейках C1, D1.

Замечания. 1. В блочном условном операторе блок ветви «нет» вместе с ключевым словом **Else** может быть опущен. Например, для оп-

ределения количества Kol вычисляемых вещественных корней уравнения из примера 2 подойдет следующий фрагмент программы

```
Kol = 0
If D >= 0 Then
    Kol = 2
End If
```

2. Когда блок состоит как здесь из одного оператора, уместно использовать не блочный условный оператор, а «*строчный*» If:

```
Kol = 0 :      If D >= 0 Then Kol = 2
```

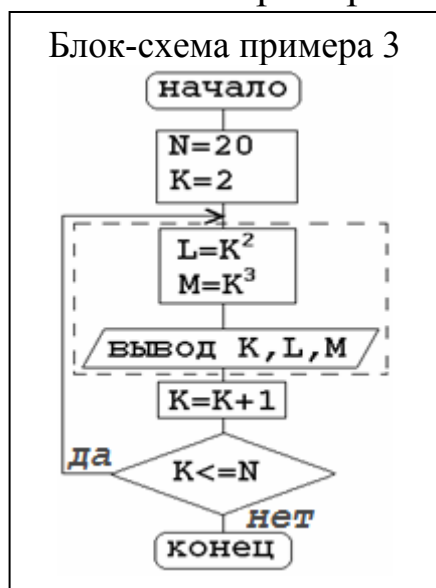
В случае справедливости условия выполнится присваивание $Kol = 2$, при невыполнении условия - следующий по порядку оператор.

Циклические вычислительные процессы

Пример 3. Вывести таблицу квадратов и кубов натуральных чисел от 2 до 20. Как результат в ячейках листа (и в окне отладки) должны появиться строки с числами:

2	4	8
...
20	400	8000

Алгоритм решения примера 3 представлен в виде блок-схемы примера 3 и макроса:



```
Sub NKvKub ( )
```

```
    K = 2:    N = 20
```

```
Start: L = K ^ 2
```

```
    M = K ^ 3
```

```
    Cells(K,1) = K: Cells(K,2) = L: Cells(K,3) = M
```

```
    Debug.Print K, L, M      ' вывод K, L, M
```

```
    K = K + 1
```

```
    If K <= N Then GoTo Start
```

```
End Sub
```

В данном алгоритме переменной K, предназначенной для хранения натуральных чисел от 2 до 20, сначала присваивается начальное значение 2. После вычисления и вывода квад-

рата и куба числа 2 в операторе присваивания $K=K+1$ к исходному значению K прибавляется 1 и сумма (3) присваивается этой же переменной, то есть записывается в ту же ячейку памяти, затирая предыдущее значение (2). После этого можно снова выполнять операторы вычисления и вывода - ведь они оперируют с содержимым всё той же ячейки, которое теперь уже увеличилось на единицу. Поэтому в «строчном» операторе If после Then записан **оператор безусловного перехода GoTo «метка»**, передающий управление на оператор $L=K^2$, отмеченный **меткой** Start. Но повторять вычисления и вывод нужно пока значение K не превысит предельной величины $N=20$, поэтому и использован условный переход по ветви «да».

Приведённый выше алгоритм реализует **циклический вычислительный процесс (цикл)**. Для цикла характерны повторения обведённых на блок-схеме пунктиром операторов, составляющих **тело цикла**. Для этого цикла характерно изменение *как в арифметической прогрессии* значения переменной K - **счетчика цикла**, определяющей число выполненных повторений, от **начального значения** (здесь от 2) до своего **конечного значения** (здесь до 20) **с шагом** (здесь 1).

Поскольку циклы широко используются при программировании, в Бейсике есть специальные операторы для организации циклов.

Для организации цикла примера 3 подойдут операторы For и Next, и с ними макрос NKvKub примет следующий вид:

```
Sub NKvKub ( )
  N=20
  For K = 2 To N Step 1      ' Для K от 2 до N с шагом 1 :
    L = K ^ 2                ' Тело цикла (начало)
    M = K ^ 3
    Cells(K,1) = K: Cells(K,2) = L: Cells(K,3) = M
    Debug.Print K, L, M      ' Тело цикла (конец)
  Next K                     ' Следующее K
End Sub
```

В приведённом макросе оператор For начала цикла организует повторное выполнение тела цикла при значениях счётчика цикла K рав-

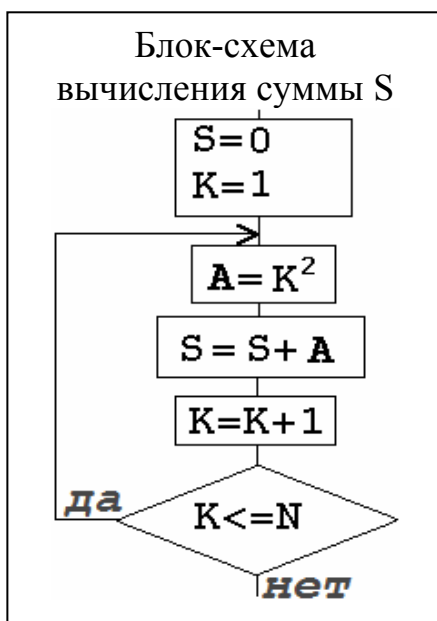
ного 2, 3, ..., N, т. е. изменяющегося от 2 до N с шагом 1. Оператор **Next** отмечает конец тела цикла. В этом операторе не обязательно указывать имя переменной-счётчика цикла. Если шаг равен (как здесь) единице, то в операторе **For** его вместе со словом **Step** можно опустить. После выхода из цикла счётчик цикла равен очередному значению, при котором тело перестанет выполняться. При шаге, не позволяющем счётчику цикла достичь предельного значения, тело цикла обходится – не выполняется ни разу.

Использование циклов при накоплении сумм

Пример 4. Для заданного значения N вычислить сумму квадратов натуральных чисел от 1 до N и проверить справедливость формулы:
 $1^2 + 2^2 + 3^2 + \dots + N^2 = N \cdot (N + 1) \cdot (2N + 1) / 6$.

Для указанной проверки вычислить сумму, стоящих слева от знака равенства слагаемых, то есть $S = 1^2 + 2^2 + 3^2 + \dots + N^2 = \sum_{k=1}^N k^2$, а также значение (обозначенного как C) выражения, стоящего справа от знака равенства, и сравнить их.

Вычисление суммы чисел, указанных слева от знака равенства, можно организовать в цикле со счётчиком цикла K, принимающим значения от 1 до N с шагом 1. При каждом значении K можно вычислять одно слагаемое $A = K^2$ и добавлять его к сумме S, которая сначала должна равняться нулю. Алгоритм вычисления суммы S представлен в виде блок-схемы и текста макроса.



Sub Sum()

N = 3 ' Назначить какое-либо значение.

S = 0 ' Подготовка к накоплению суммы.

For K = 1 **To** N **Step** 1

A = K ^ 2

S = S + A ' Пополнение суммы.

Next K

C = N * (N + 1) * (2 * N + 1) / 6

Debug.Print N, S, C

End Sub

При выполнении макроса в режиме отладки полезно проследить (наводя курсор на переменную S) то, как пополняется сумма при каждом выполнении оператора $S = S + A$. Для принятого выше значения $N=3$ верно вычисленная сумма должна составить 14. Такое же значение S даёт вычисление по формуле. Совпадут ли результаты при иных значениях N?

Пример 5. Значения высоты H и радиуса R из примера 1 заданы начиная с ячеек A2 и B2 в нескольких последовательных строках соответственно столбцов A и B. Требуется: 1) заполнить столбец C, начиная с ячейки C2, значениями объёмов конусов, 2) записать в ячейки D2, E2 и F2 соответственно общий объём всех конусов, количество конусов и средний объём конуса.

Для перебора строк (начиная со второй) и записи вычисляемых значений объёмов уместно применить цикл, однако в общем случае не известно в какой строке оканчиваются данные. Поэтому обработка данных должна вестись до тех пор, пока есть данные в строках. Циклы, которые организуют повторное выполнение тела цикла пока справедливо (или наоборот не справедливо) некоторое условие, называют *итерационными циклами*. В VBA такие циклы начинается оператор **Do** (выполнять), за которым могут следовать опции: служебные слова **While** (пока справедливо) или **Until** (пока не справедливо) и *условие*. Конец тела цикла отмечает оператор **Loop** (петля, цикл), который также может иметь названные опции (только если они не указаны вместе с оператором **Do**).

Решение 1-ой части примера 5 даёт следующий макрос.

```
Sub Primer5_1( )
```

```
  i = 2
```

```
Do While Cells(i, 1) <> Empty ' Цикл «Выполнять пока справедливо»
```

```
    H = Cells(i, 1): R = Cells(i, 2)      ' Назначить конкретные значения H и R
```

```
    V = H/3*3.14 * R^2: Cells(i, 3)=V      ' Вычислить объём конуса
```

```
    i = i + 1
```

```
Loop
```

```
End Sub
```

В условии оператора цикла **Do While** проверяется: не равно ли содержимое очередной *i*-ой ячейки 1-го столбца (столбца А) значению Empty («пусто»)? И пока оно не «пусто» вычисляется объём и увеличивается каждый раз на единицу номер *i*, указывающий на очередную строку. Цикл **Do Until** представлен во 2-ой части задачи.

Для решения 2-ой части задачи надо дополнить приведённый макрос накоплением суммы *S* и количества *k* обработанных строк:

Sub Primer5_2()

S=0: k = 0

Do Until Cells(k+2, 1) = Empty ' Цикл «Выполнять пока не справедливо»

H = Cells(k+2, 1): R = Cells(k+2, 2) ' Назначить конкретные значения H и R

V= H/3*3.14 * R^2: Cells(k+2, 3)=V ' Вычислить объём конуса

S = S +V: k = k + 1 ' Накапливание суммы и количества

Loop

Cells(2, 4) = S: Cells(2, 5) = k: Cells(2, 6) = S / k ' Результаты- в D2, E2, F2

End Sub

Можно отметить, что количество *k* накапливается как сумма *S*, но в качестве слагаемых *k* накапливаемому количеству добавляются единицы. В макросе номер обрабатываемой строки получается увеличением количества *k* на 2.

Пример 6. По списку повторяющихся фамилий абонентов и сумм (счетов см. рис. 6.2) за каждый телефонный разговор абонента (записанных соответственно в столбцах В и С, начиная с ячеек В3 и С3) вычислить для указанной фамилии абонента: общую сумму к оплате, количество его звонков, среднюю стоимость его звонка.

□ **Макрос - Решение примера 6**

Sub Primer6() ' == Образец задания 1 ==

Fam\$="Котов": i = 3: S=0: Kol = 0:

Do While Cells(i, 2) <> Empty

If Cells(i, 2) = Fam\$ **Then** ' Вычисляя для Fam\$

S=S+ Cells(i, 3) : Kol = Kol +1

End If

	А	В	С
1			
2		ФИО	Счёт
3		Котов	15,36
4		Гудов	18,33
5		Васин	8,31
6		Ежов	14,17
7		Васин	25,84
8		Котов	11,83
9		Ежов	16,33

Рис. 6.2.

$i = i + 1$ ' i – номер обрабатываемой строки

Loop

' Запись результатов в B1, D1 и F1:

Cells(1, 2) = S: Cells(1, 4) = Kol: Cells(1, 6) = S / Kol

End Sub

Следует отметить, что в условном блочном операторе, изменение счётчика Kol и суммирование производится лишь тогда, когда содержимое ячейки совпадает с содержимым текстовой переменной Fam\$. Здесь текстовая (или строковая – типа String) переменная Fam\$ по старым правилам Бейсика отмечена знаком \$ на конце. Перед циклом ей присваивается значение текстовой константы с фамилией (здесь "Котов"). Можно настроить макрос на считывание фамилии из фиксированной ячейки, скажем, из ячейки \$H\$1 по оператору Fam\$ = Cells(1,8). Тогда перед запуском макроса в эту ячейку надо скопировать требуемую фамилию. В разделе «3.Дополнительные сведения» этот вопрос решён в общем виде – ячейки с фамилиями в процессе анализа просто указываются (выделяются) мышкой.

2.Задание 1 к лабораторной работе

Подготовить в соответствии с вариантом задания таблицу со списком наименований (фамилий) и значений необходимых полей. Подготовить макрос, в котором для указанного (среди его операторов) наименования (фамилии) вычислить и записать в ячейки значения требуемых параметров.

Предъявить работающий макрос преподавателю. Текст макроса и схематичный вид результатов его выполнения записать в отчёт.

Варианты заданий

1. В списке (с колонками: наименование товара, число упаковок, вес упаковки) товаров поступивших на склад за месяц по наименованию указанного товара определить: общее число упаковок и общий вес данного товара на складе.
2. В таблице заказов для продукта, заданного своим наименованием, определить: общую потребность (объём требуемого продукта) и число заказов (заявок) на данный продукт.

3. В таблице-распечатке телефонных вызовов по номеру абонента определить: число его исходящих звонков, общую продолжительность разговоров и среднюю продолжительность разговора.
4. В библиотечном списке методической литературы по названию учебной дисциплины определить: число методичек и средний объем (в страницах) одной методички по указанной дисциплине.
5. В таблице с аттестационными оценками школьников по пройденным предметам определить: среднюю оценку выбранного школьника и число предметов с аттестацией на «5».
6. В таблице с аттестационными оценками школьников по пройденным предметам определить: среднюю оценку по выбранному предмету и число отличником по данному предмету.
7. В списке поставок (за неделю) товаров со склада в магазины определить: число магазинов, в которые поставляется указанный товар, а также общий объем поставляемого товара.
8. По списку повторяющихся фамилий сотрудников и часов сверхурочной работы (в указанные дни) определить количество дней и общую сумму часов сверхурочной работы сотрудника с фамилией, указанной в этом списке.
9. Для указанной фамилии в месячной распечатке (по дням) сотрудников, опоздавших на работу, определить число дней с опозданиями и общее время опозданий.
10. В списке с названиями фирм и количеств легковых автомобилей этих фирм, проехавших по участку дороги в каждый из нескольких дней недели определить: процент всех автомобилей выбранной фирмы от общего числа автомобилей.
11. В списке (с колонками: наименование товара, вес, сорт) товаров, поступивших на склад за месяц по наименованию указанного товара данного сорта определить: общий вес указанного товара данного сорта (1-го или 2-го).
12. Для указанной фамилии в списке (по дням месяца) водителей автопредприятия, оштрафованных ГИБДД, определить: общее число нарушений правил и общую сумму штрафов.

13. Для указанной фамилии в списке спонсорских пожертвований (за год) определить: число пожертвований и среднюю сумму одного пожертвования.
14. Для указанной фамилии в распечатке (по дням) вложений (вкладов) на общий банковский счёт определить: общую сумму вкладов и среднюю сумму одного вклада.
15. В списке (по матчам сезона) хоккеистов, имеющих удаления в матчах, для выбранного хоккеиста определить: общее время удалений и общее число матчей с удалениями.

3.Дополнительные сведения

Взаимодействие макроса с объектами Excel

Макрос обычно взаимодействует с объектами приложения, в котором он определён - host-приложения (host – хозяин). Объектами Excel являются *Книга (Workbook)*, *Рабочий лист (WorkSheet)* с *Диапазонами ячеек (Range)*. Объекты обладают свойствами, имеют методы, позволяющие изменять свойства. Объекты могут реагировать на происходящие события. Такую возможность целесообразно использовать для эффективного решения **примера 6**, в котором по очередной указанной (выделенной) ячейке с фамилией должны вычисляться суммарные показатели для всех строк с этой фамилией.

Для получения адреса указанной (выделенной) ячейки важна реакция листа на событие «Изменение выделения».

Объекты Excel представлены в окне обозревателя проекта (Project) среды VBA. Клик по значку *Лист1* в окне Project настраивает окно кода: *Лист1 (code)*. А выбор на этом окне в (General) WorkSheet (после щелчка по треугольнику справа) приводит к генерированию операторов заголовка и конца процедуры обработки события «Изменение выделения»:

```
Private Sub Worksheet_SelectionChange(ByVal Target As Range)
```

```
End Sub
```

В ответ на изменение выделения диапазона (ячейки) Excel передаёт как значение (**By Value**), т.е. – копирует, в объект Target (мишень) этой внутренней (private – личная) процедуры объект типа Range (диапа-

зон) с характеристиками выделенной ячейки (например, адресом: Target.Address). Вообще говоря, все действия, связанные с обработкой этой ячейки, можно поместить внутрь данной процедуры. Но обычно общие действия для ряда объектов (в частности для различных листов) выполняют в модулях. Следуя этому принципу можно всю обработку ячейки из макроса примера 6 оформить как процедуру, в которую из процедуры обработки события передаётся адрес Addr только что выделенной ячейки. В таком случае процедуру в модуле (названную, например, ModuleSub) следует начать оператором

Sub ModuleSub(ByVal Addr As String) 'Addr – текстовая строка под адрес
Для её вызова и передачи в неё фактического адреса только что выделенной ячейки в процедуре обработки события помещают **оператор вызова процедуры Call** :

Call ModuleSub(Target.Address) ' Вызов процедуры с передачей в неё адреса

Теперь можно сформулировать

сценарий универсального решения примера 6:

1. По сочетанию клавиш Ctrl+Shift+R макросом запускается процесс анализа общих сумм и количества звонков абонентов.
2. Выделяется ячейка. Если она пуста, то следует переход к пункту 4, иначе (ячейка содержит фамилию) выполняется пункт 3.
3. Процедура обработки события «Изменение выделения» передаёт адрес только что выделенной ячейки в процедуру ModuleSub, которая вычисляет и выводит требуемые величины. Затем следует переход к пункту 2, где ожидается выделение очередной ячейки.
4. После выделения пустой ячейки процесс анализа прекращается.

В макросе RunMac для запуска анализа можно использовать общую (public) для всех процедур переменную, например, переменную с именем OnOff. Сначала она имеет значение Empty, а потом OnOff=1, когда процесс анализа запускается («включается») макросом:

Public OnOff ' объявление общей переменной

Sub RunMac()

OnOff = 1: Cells(2, 1) = "Вкл" ' Символы Вкл - в ячейку \$A\$2

End Sub

В процедуре (рабочего листа) обработки события «Изменение выделения» вызов ModuleSub выполняется лишь тогда, когда OnOff =1:

Private Sub Worksheet_SelectionChange(**ByVal** Target **As** Range)

If OnOff = 1 **Then** **Call** ModuleSub(Target.Address) ' Вызов с передачей адреса
End Sub

При завершении анализа, когда при OnOff =1 адрес указывает на пустую ячейку, необходимо в процедуре ModuleSub назначить OnOff = Empty, «выключить» анализ и выйти из процедуры по оператору выхода **Exit Sub**. С этим добавлением размещаемая в том же модуле, что и макрос процедура ModuleSub имеет следующий вид:

Sub ModuleSub(**ByVal** Addr **As** String) 'Addr – текстовая строка под адрес

If Range(Addr) = Empty **Then**

OnOff = Empty: Cells(2, 1) = "Выкл": **Exit Sub** ' Выкл - в ячейку \$A\$2

End If

Fam\$ = Range(Addr) ' Копировать фамилию в Fam\$

i = 3: S = 0: Kol = 0 ' Текст из примера 6:

Do While Cells(i, 2) <> Empty

If Cells(i, 2) = Fam\$ **Then**

S = S + Cells(i, 3)

Kol = Kol + 1

End If

i = i + 1

Loop

Cells(1, 2) = S: Cells(1, 4) = Kol ' Запись результатов в B1, D1 и F1:

If (Kol <> 0) **Then** Cells(1, 6) = S / Kol ' Kol=0 при некорректном указании

Cells(1, 8) = Fam\$ ' Фамилию - в ячейку \$H\$1

End Sub

Вывод результатов в форму можно обеспечить, если использовать **объект UserForm** (экранная форма или просто форма), который составляет часть пользовательского интерфейса приложений Windows. Объекту UserForm (как экземпляру класса UserForm) присущи его свойства и методы. Каждый объект UserForm включает в себя также модуль класса, в который можно добавлять собственные методы и

свойства или код для обработки событий формы. В приложениях MS Office, и в частности в Excel, форма подключается после выбора *UserForm* в пункте *Insert* меню редактора VBA.

В результате появляется пустое диалоговое окно формы, **имя формы** UserForm1 включается в Project, в окне Properties представлены **свойства формы**, доступные для редактирования. Можно изменить, например, имя (Name) формы на новое frmITOG, а также свойство Caption (название), набив, например, слово ИТОГ в поле справа от Caption. После чего это новое название появится в заголовке формы (см. рис. 6.3).

К объекту UserForm можно добавлять элементы управления с панели Toolbox.

Для наших целей подойдёт Label (Надпись) – пиктограмма с буквой «А», которую следует отбуксировать в центр формы. После этого надпись Label1 становится подобъектом формы frmITOG; она имеет своё свойство Caption (записывается как frmITOG.Label1.Caption), которое можно использовать для вывода результатов в **управляющий элемент Надпись**. Для примера 6 в конец процедуры ModuleSub (перед оператором End Sub) добавлены следующие операторы:

```
frmITOG.Caption = Fam$ & "Сум.СЧЕТ:" ' вывод на место названия формы
frmITOG.Label1.Caption = S & " руб." ' вывод в Надпись
Call frmITOG.Show ' вызов метода Show делает форму видимой
```

По сочетанию клавиш Ctrl+Shift+R можно снова запускать теперь уже итоговый макрос для примера 6.

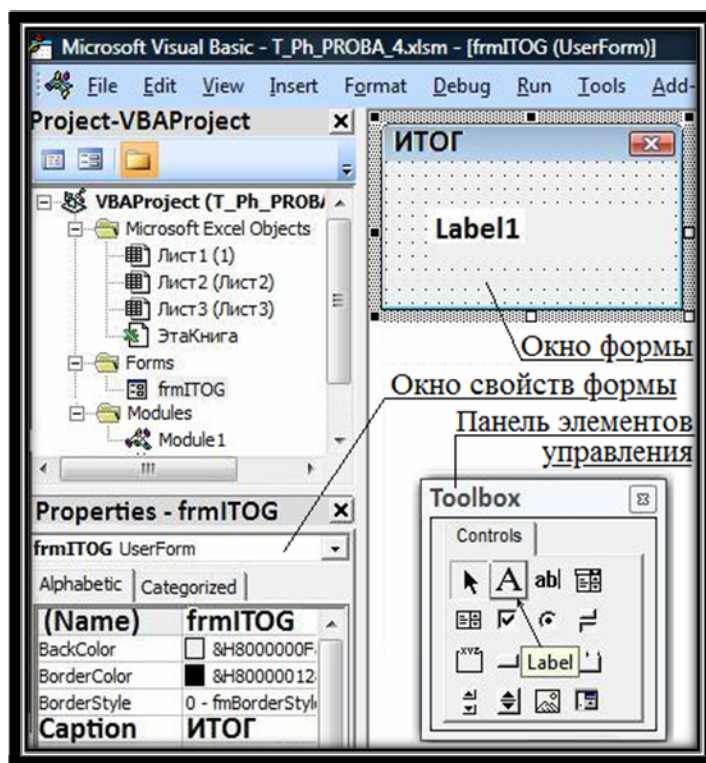


Рис. 6. 3.

После запуска в ячейку \$A\$2 выводятся символы «Вкл» - включен анализ, и можно выделять ячейку с фамилией. После её выделения процедура модуля выполняет вычисление общей суммы, определя-

	A	B	C	D	E	F
1	Сумма =	27,19	Кол-во=	2	Стоим.1=	13,595
2	Вкл	ФИО	Счёт			
3		Котов	15,36			
4		Гудов	18,33			
5		Васин	8,31			
6		Ежов	14,17			
7		Васин	25,84			
8		Котов	11,83			
9		Ежов	16,33			
10						

Рис. 6. 4.

ет количество вызовов и среднюю стоимость одного звонка указанного абонента, а также заполнение результатами ячеек B1, D1 и F1 (см. рис. 6.4). Фамилия выводится в H1. Кроме того будет выведена форма с фамилией и итоговой суммой. Закрыв форму, можно снова выделить ячейку с фамилией и получить суммарные показатели для этой фамилии. После выделения пустой ячейки процесс анализа завершается, а в ячейку \$A\$2 выводятся символы «Выкл» - анализ выключен.

4.Задание 2 к лабораторной работе

Дополнить и изменить макрос 1-ой части так, чтобы по выделенной ячейке таблицы с наименованием (фамилией) этот макрос вычислял и записывал значения требуемых параметров.

Предъявить работающий макрос преподавателю. Текст макроса и схематичный вид результатов его выполнения записать в отчёт.

Рекомендуемая литература

1. Информатика базовый курс. Учебник под редакцией С.В. Симоновича – С.Пб.: «Питер», 2005.
2. Информатика и информационные технологии, под ред. Романовой Ю.Д. - М: Издательство «ЭКСМО», 2010.
3. Алехина Г.В., Основы информатики – М.: Маркет ДС, 2009.
4. Дж. Кокс, С. Ламберт, К. Мюррей, К. Фрай, Дж. Преппернау, Microsoft Office System 2007. Русская версия - С.Пб.: ЭКОМ, 2007.
5. Л. Д. Слепцова, Программирование на VBA в Microsoft Office 2007 – М.: Диалектика, 2007.

СОДЕРЖАНИЕ

Введение.....	3
Лабораторная работа №1: Операционная система Windows.	4
Лабораторная работа №2: Работа с текстовым процессором MS Word.....	15
Лабораторная работа №3: Работа с табличным процессором Microsoft Excel 2007.....	30
Лабораторная работа №4: Система управления базой данных (СУБД) Access. Создание таблиц и схемы данных.....	42
Лабораторная работа №5: Анализ информации таблиц и БД Access. Создание запросов.	54
Лабораторная работа №6: Начала алгоритмизации. Обработка в Excel табличных данных с использованием макросов на VBA.....	66
Рекомендуемая литература	83

Учебное издание

Калядин Вячеслав Иванович, Макаров Алексей Иванович

**Основы работы на персональном компьютере.
Сборник лабораторных работ по дисциплине «Информатика».**

Под редакцией авторов

*Оригинал-макет подготовлен редакционно-издательским отделом
МГТУ «МАМИ»*

По тематическому плану внутривузовских изданий учебной литературы на 2010 г.

Подписано в печать 24.06.2010 Формат 90х60 1/16 Бумага 80 г/м²
Гарнитура «Таймс». Ризография. Уч. печ. л. 5,3
Тираж 550 экз Заказ № 112-10

МГТУ «МАМИ»

107023, г. Москва, Б.Семёновская ул. 38.