

Федеральное агентство по образованию

САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО

---

*В.С. ТУТЫГИН*

**ИНФОРМАТИКА**  
**MICROSOFT OFFICE 2016**

Учебное пособие

Санкт-Петербург  
Издательство Политехнического университета  
2016

УДК 681.3.06 (075.8)

ББК 32.81 м73

Т 916

Тутыгин В.С. Информатика. Microsoft Office 2016.: Учеб. пособие. СПб.: Изд-во Политехн. ун-та, 2016. 216 с.

Пособие соответствует основным разделам дисциплины «Информатика» направлений бакалаврской подготовки 09.03.01 и 09.03.04.

В пособии изложены основные сведения о профессиональных приемах технологии подготовки текстовых документов, графического материала, таблиц, создания баз данных, конструирования презентаций средствами пакета Office 2016.

Описаны основы создания приложений пользователя в среде Excel 2016 и Access 2016 на языке Visual Basic for Application.

Приведены задания для упражнений и примеры их выполнения.

На Yandex-диске <https://yadi.sk/d/SR31asX6oYxHx> приведены файлы с текстами, рисунками, таблицами, базами данных для выполнения упражнений, примеры выполненных студентами самостоятельных работ по всем темам, а также краткое изложение материала дисциплины, представленное в форме презентаций в PowerPoint 2016.

Предназначено для студентов, проходящих подготовку по направлениям 09.03.01 «Информатика и вычислительная техника», 09.03.04 «Компьютерная инженерия». Оно может также быть использовано студентами, обучающимися по другим направлениям, изучающими дисциплину «Информатика», и преподавателями.

Табл. 34. Ил. 82. , Библиогр.: 5 назв.

Печатается по решению редакционно – издательского совета Санкт – Петербургского государственного политехнического университета

© Тутыгин В.С., 2016

© Санкт – Петербургский политехнический университет  
Петра Великого, 2016

ISBN \_\_\_\_\_

## **Оглавление**

<b>ВВЕДЕНИЕ.....</b>	<b>8</b>
<b>1. MICROSOFT WORD 2016 .....</b>	<b>9</b>
1.1. ЗАПУСК РЕДАКТОРА WORD .....	9
1.2. Создание простых текстовых документов.....	11
Форматирование страницы.....	12
Форматирование абзаца .....	13
Форматирование текста.....	13
Использование автотекста .....	14
Вставка формул.....	15
Вставка иллюстраций .....	16
Вставка таблиц.....	17
Вставка фигур .....	17
Проверка правописания.....	19
1.3. РЕДАКТИРОВАНИЕ ТЕКСТА.....	19
Пометка текста.....	19
Вставка, копирование, перемещение фрагмента текста .....	20
Изменение шрифта, стиля написания, размера символов .....	21
Копирование формата.....	21
Отмена последней операции редактирования.....	21
Загрузка текстового файла в окно редактирования .....	22
1.4. ОБРАБОТКА СЛОЖНОГО ДОКУМЕНТА.....	22
Создание титульной страницы.....	22
Структурирование документа. Использование шаблонов и стилей.....	23
Введение колонтитулов.....	27
Нумерация страниц.....	27
Формирование сносок .....	28
Маркировка и нумерация абзацев .....	28
Обрамление и заливка абзацев .....	29
Многоколоночное расположение текста .....	29
Автонумерация рисунков.....	31
Вставка сложных формул.....	31
Создание сложной таблицы.....	32
Построение и вставка графиков .....	32
Автоформирование оглавления.....	33
Автоформирование списка литературы.....	35
Создание списка иллюстраций.....	35
Создание предметного указателя .....	35
ИМПОРТ ОБЪЕКТОВ ИЗ ДРУГИХ WINDOWS - ПРИЛОЖЕНИЙ .....	36
Создание объекта и вставка .....	36

<i>Вставка текста из файла</i> .....	37
<i>Редактирование формата рисунков</i> .....	37
1.5. Создание прикладных документов .....	37
<i>Создание таблиц с вычисляемыми полями и полями со списком</i> .....	37
<i>Создание документов путем слияния</i> .....	42
1.6. Создание гиперссылок.....	46
1.7. Сохранение документа на магнитном диске .....	46
1.8. Печать документа.....	47
1.9. Выход из редактора WORD .....	47
Приложение. Клавиши быстрого доступа .....	48
1.10. Упражнения.....	49
<b>2. EXCEL 2016 .....</b>	<b>54</b>
2.1. Запуск EXCEL 2016 .....	54
2.2. Создание таблицы .....	56
<i>Ввод текста, цифровых данных и формул в клетку (ячейку) таблицы</i> .....	56
<i>Автозаполнение</i> .....	57
<i>Автосуммирование</i> .....	60
2.3. Редактирование таблицы .....	60
<i>Пометка элементов таблицы</i> .....	61
<i>Редактирование ячейки</i> .....	61
<i>Вставка, копирование, перемещение фрагмента таблицы</i> .....	62
<i>Форматирование таблицы</i> .....	63
<i>Отмена последней операции редактирования</i> .....	66
<i>Загрузка файла в окно редактирования</i> .....	66
2.4. Создание и редактирование диаграмм.....	66
<i>Создание диаграмм</i> .....	66
<i>Редактирование и анализ диаграмм</i> .....	68
2.5. Сохранение документа на магнитном диске .....	68
2.6. Подготовка к печати.....	69
2.7. Вывод на печать .....	69
2.8. Базы данных .....	70
<i>Сортировка</i> .....	70
<i>Поиск в базе данных по нескольким критериям</i> .....	70
2.9. Анализ данных .....	72
2.10. Работа со сводными таблицами .....	74
2.11. Решение задач .....	76
<i>Подбор параметра</i> .....	76
<i>Поиск оптимальных решений</i> .....	77
2.12. Создание документов на основе данных в таблицах .....	80
2.13. Работа с макросами .....	81

<i>Создание макроса</i> .....	82
<i>Запуск макроса</i> .....	82
<i>Создание средств запуска макросов</i> .....	82
2.14. ВЫХОД ИЗ EXCEL 2016 .....	83
ПРИЛОЖЕНИЕ 1. КЛАВИШИ БЫСТРОГО ДОСТУПА .....	83
ПРИЛОЖЕНИЕ 2. КЛАВИШИ БЫСТРОГО ПЕРЕМЕЩЕНИЯ ПО РАБОЧЕМУ ЛИСТУ И КНИГЕ.....	84
2.15. УПРАЖНЕНИЯ.....	85
<b>3. VISUAL BASIC FOR APPLICATIONS.....</b>	<b>97</b>
3.1. ОСНОВНЫЕ СРЕДСТВА VBA В EXCEL 2016 .....	97
3.2. ПРИМЕНЕНИЕ VBA.....	99
3.3. ОТКРЫТИЕ ОКНА ДЛЯ РАЗРАБОТКИ ПРОГРАММ НА VBA.....	99
<i>Создание модуля</i> .....	100
<i>Создание UserForm.</i> .....	101
<i>Панель элементов VBA</i> .....	102
<i>Запуск приложения из UserForm</i> .....	106
3.4. Создание СРЕДСТВ ЗАПУСКА ПРИЛОЖЕНИЯ VBA ИЗ WINDOWS .....	107
3.5. УПРАЖНЕНИЯ 1 .....	107
3.6. ОБРАБОТКА И ПЕРЕХВАТ ОШИБОК В ПРОГРАММАХ НА VBA .....	112
3.7. УПРАЖНЕНИЯ 2 .....	115
3.8. ИСПОЛЬЗОВАНИЕ МАКРОСОВ.....	118
3.9. ЗАЩИТА ПРОГРАММНОГО КОДА ПРИЛОЖЕНИЙ ПОЛЬЗОВАТЕЛЯ.....	120
3.10. ПРИМЕРЫ ВЫПОЛНЕНИЯ ЗАДАНИЙ .....	121
ПРИЛОЖЕНИЕ 1. ПРИМЕРЫ МЕТОДОВ И СВОЙСТВ ОБЪЕКТОВ VBA В EXCEL.....	134
ПРИЛОЖЕНИЕ 2. ПРИМЕРЫ МЕТОДОВ, СВОЙСТВ И СОБЫТИЙ ЭЛЕМЕНТОВ УПРАВЛЕНИЯ VBA В EXCEL .....	135
ПРИЛОЖЕНИЕ 3. ОСНОВНЫЕ ОПЕРАТОРЫ ЯЗЫКА VBA .....	137
<b>4. ACCESS 2016 .....</b>	<b>141</b>
4.1. ОБЪЕКТЫ ACCESS .....	141
4.2. Создание и сохранение файла базы данных .....	145
4.3. Создание таблицы.....	145
4.4. Создание и использование форм .....	147
<i>Создание формы</i> .....	147
<i>Поиск данных в формах</i> .....	148
<i>Сортировка данных (по возрастанию/убыванию)</i> .....	148
<i>Фильтрация данных (фильтр по выделенному)</i> .....	148
<i>Фильтрация данных (фильтр по форме)</i> .....	149
<i>Фильтрация и сортировка данных</i> .....	149
<i>Создание запросов к однотабличной базе данных</i> .....	149
4.5. Создание реляционной БД и работа с ней .....	149

<i>Формирование структуры связей между таблицами</i> .....	150
<i>Обеспечение целостности данных в БД</i> .....	151
<i>Нормализация таблиц</i> .....	151
<i>Роль связей между таблицами. Установление целостности данных.</i> ...	152
4.6. ТЕХНОЛОГИЯ ЗАГРУЗКИ ТАБЛИЦ БД С ИСПОЛЬЗОВАНИЕМ ФОРМ .....	153
4.7. Создание реляционной БД.....	154
<i>Создание таблиц БД</i> .....	154
<i>Формирование структуры связей между таблицами</i> .....	155
<i>Создание форм</i> .....	156
<i>Создание поля со списком в формах</i> .....	156
<i>Создание поля со списком в окнах подчиненных форм</i> .....	157
4.8. Создание отчетов.....	157
4.9. Создание кнопочных форм управления в БД .....	158
4.10. ПАРАМЕТРЫ ЗАПУСКА .....	159
4.12. МАКРОСЫ.....	167
<i>Создание макроса</i> .....	169
<i>Установка автоматического запуска макроса при открытии БД</i> .....	169
<i>Создание кнопки для запуска макроса</i> .....	169
<i>Установка режима запуска при возникновении события</i> .....	170
<i>Создание и использование макросов с условием</i> .....	171
4.13. ЗАПРОСЫ .....	175
<i>Назначение запросов</i> .....	175
<i>Основные виды запросов</i> .....	175
<i>Средства создания запросов</i> .....	175
<i>Вычисляемые поля в запросах на выборку</i> .....	176
<i>Создание запросов с параметрами</i> .....	177
<i>Создание запросов с использованием групповых операций</i> .....	178
<i>Запросы на обновление, добавление и удаления записей в таблицах БД..</i>	180
4.14. ЗАПРОСЫ НА ЯЗЫКЕ SQL .....	182
<i>Запрос на выборку к одной таблице</i> .....	183
<i>Запросы на выборку к системе связанных таблиц</i> .....	184
4.15. КОНСТРУИРОВАНИЕ СЛОЖНЫХ ОТЧЕТОВ .....	186
4.16. РАЗРАБОТКА ПРИЛОЖЕНИЙ ПОЛЬЗОВАТЕЛЯ НА VBA В СРЕДЕ ACCESS ....	190
<i>Основные средства VBA в Access</i> .....	191
<i>Применение VBA в Access</i> .....	193
<i>Открытие окна для разработки программ на VBA</i> .....	193
<i>Создание модуля</i> .....	194
<i>Программирование форм и отчетов</i> .....	195
<i>Пример Access-приложений, разработанных с использованием VBA</i> .....	196
4.17. ЗАЩИТА БАЗ ДАННЫХ.....	201
<i>Защита базы данных (accdb-файла) с помощью пароля</i> .....	201

<i>Сокрытие объектов и окна базы данных .....</i>	202
<i>Защита путем создания файла ACCDE базы данных .....</i>	203
<b>5.POWERPOINT 2016 .....</b>	<b>204</b>
5.1. Создание титульного слайда и файла презентации .....	204
5.2. Добавление слайда в состав презентации .....	205
5.3. Создание слайда .....	205
<i>Создание объектов в слайдах.....</i>	205
<i>Вставка текстового фрагмента из документа Microsoft Word .....</i>	206
<i>Вставка диаграммы PowerPoint.....</i>	206
<i>Вставка диаграммы Microsoft Excel 2016 .....</i>	207
<i>Вставка таблицы Microsoft Excel 2016.....</i>	207
<i>Вставка объектов WordArt .....</i>	207
<i>Вставка рисунков .....</i>	208
<i>Вставка видео- и звуковых клипов.....</i>	208
<i>Вставка объектов, созданных в других приложениях Windows .....</i>	208
5.4. Добавление слайда из файла.....	209
5.5. Сортировка слайдов .....	209
5.6. Редактирование презентации .....	209
5.7. Создание и настройка анимации .....	210
5.8. Установка режима просмотра презентаций.....	210
5.9. Просмотр презентаций .....	211
5.10. Изменение последовательности просмотра слайдов .....	211
5.11. Создание и использование скрытых слайдов.....	212
5.12. Создание и использование при демонстрации презентации кнопок управления.....	212
5.13. Добавление к презентации раздаточного материала .....	213
Упаковка презентаций .....	214
<b>СПИСОК ЛИТЕРАТУРЫ .....</b>	<b>215</b>

## Введение

Пакет программ Microsoft Office Professional 2016, важнейшими из которых являются программы Word, Excel, Access, PowerPoint, широко используется для организации компьютерного делопроизводства, при оформлении учетно – отчетной документации, статей, презентаций, является общепринятым средством для оформления студентами отчетов по лабораторным работам, курсовых и дипломных работ.

Освоение всех возможностей и приемов профессиональной работы с программами пакета Microsoft Office Professional 2016 по имеющейся литературе и мультимедийным учебникам требует большого времени.

Учебное пособие знакомит с основными приемами профессиональной работы с пакетом программ Microsoft Office Professional 2016. На Yandex-диске <https://yadi.sk/d/SR31asX6oYxHx> приводятся текстовые и графические файлы, содержащие исходные материалы (тексты, таблицы, программы, рисунки, базы данных) для выполнения упражнений и примеры выполненных самостоятельных заданий, компьютерные тесты.

Объем, подбор и изложение материала рассчитаны на освоение приведенного материала при объеме практических занятий 68 часов.

Приведенные в учебном пособии упражнения рассчитаны на знакомство обучаемых с основами работы в среде Windows, в частности, с приемами пометки объектов, вызовом контекстного меню и использованием буфера обмена.

Для компактности изложения описание действий пользователя в программах Microsoft Word2016, Excel 2016, Access2016, PowerPoint2016 даны кратко и только одним, но наиболее удобным способом.

## 1. Microsoft Word 2016

Текстовый процессор WORD 2016, работающий под управлением операционной системы WINDOWS 10/8/7, дает возможность создавать разнообразные документы: деловые письма, отчеты, оригинал-макеты брошюр для типографского издания, серии платежных документов, рекламные листки и т. д., по своим возможностям близок к настольным издательским системам.

При создании документа с помощью Word нужно выполнить следующие действия:

1. Установить основные параметры формата документа (задать поля, межстрочные интервалы, вид, размер и цвет шрифта, параметры колонтитулов).
2. Создать текст документа, используя, по возможности, шаблоны документов.
3. Проверить орфографию.
4. Вставить рисунки, диаграммы, таблицы .
5. Выполнить обработку документа (выполнить оформление абзацев, оформить колонтитулы, сформировать сноски, оглавление, список литературы, предметный указатель, список рисунков и таблиц, пронумеровать страницы).

### **1.1. Запуск редактора Word**

Запуск программы - редактора текстов Word в среде WINDOWS 10/8/7 можно выполнить из главного меню:

[Пуск] – [Все приложения] – [Word 2016]<sup>1</sup>

В результате откроется окно Word (Рис. 1.1).

Элементами окна Word являются:

- строка заголовка (Word);
- кнопка Файл вызова системного меню и выхода из редактора;
- лента (главное меню) с набором вкладок, на каждой из которых помещены пиктограммы, объединенные в группы;

---

<sup>1</sup> Здесь и в дальнейшем запись в виде последовательности названий пунктов текстового меню, заключенных в квадратные скобки и разделенных знаками дефис означает последовательность выбора пунктов меню, необходимая для выполнения того или иного действия.

## 1. Microsoft Word 2016

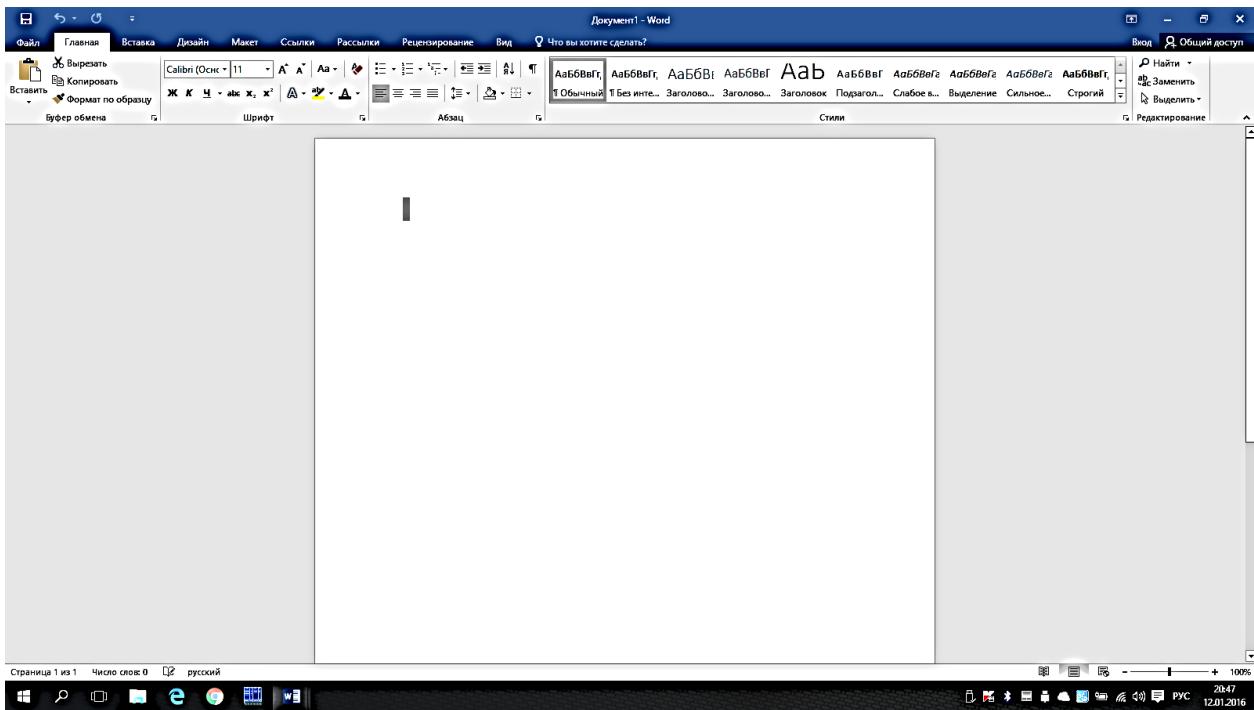


Рис. 1.1.

- панель быстрого доступа с размещенными на ней пиктограммами наиболее часто выполняемых действий: «Сохранить», «Отменить ввод» и «Повторить ввод»;
  - размерные кнопки - "Развернуть" и - "Свернуть";
  - линейки прокрутки;
  - статусная строка;
  - рабочая область окна для ввода текста;

После этого можно вводить текст с клавиатуры или загрузить в окно текстовый файл с магнитного диска. Выбор действий в Word производится с помощью главного меню, меню панели быстрого доступа, контекстных меню и меню смарт-тегов<sup>2</sup>. Основным инструментом пользователя при работе в среде WORD 2016 является графический манипулятор мышь. Выполняемая функция определяется формой, которую приобретет курсор мыши, и тем, какую клавишу мыши Вы нажимаете и каким образом.

<sup>2</sup> Смарт-тег – это пиктограмма, появляющаяся после выполнения некоторых действий. Активизация смарт-тега с помощью мыши приводит к появлению дополнительного меню режима.

Для выполнения какого-либо действия нужно открыть вкладку, найти нужную группу и активизировать пиктограмму щелчком левой кнопки мыши, например: #Вставка# – !Страницы! –[Титульная страница]<sup>3</sup>

В группу может входить пиктограмма  - «Открыть диалоговое окно». В диалоговом окне можно более детально определить выполняемое действие.

Состав групп на вкладках и пиктограмм в группах – контекстно зависим.

При пометке объектов могут автоматически всплывать контекстные меню. Контекстное меню любого помеченного объекта можно вызвать щелчком правой кнопки мыши.

Количество пиктограмм на панели быстрого доступа можно увеличить, открыв меню «Настроить панель быстрого доступа» щелчком левой кнопки мыши на пиктограмме, находящейся справа от панели. В открывшемся меню можно выбрать команды из списка или, если требуемой команды в списке нет, выбрать пункт «Другие команды...».

### Формы курсора мыши

-  курсор ожидания загрузки программы;
-  курсор маркования строки;
-  указатель выбора объекта: кнопки, участка текста, рисунка;
-  курсор маркования текста;
-  курсор изменения размеров окна ;
-  курсор копирования формата.

### 1.2. Создание простых текстовых документов

До начала набора текста можно выбрать один из стандартных шаблонов документа:

!Файл! – [Создать]

далее выбрать вид шаблона. Если стандартный шаблон документа не нужен, нужно выбрать пункт «Новый документ» в области задач.

Желательно до начала набора текста документа задать параметры формата документа: формат страницы, абзаца, текста.

---

<sup>3</sup> Значком ## здесь и далее обозначается вкладка, значком !! – группа и значком [] – пиктограмма.

## Форматирование страницы

При форматировании страницы устанавливаются:

- размеры полей (верхнего, нижнего, левого, правого), ориентация листа, параметры колонтитулов;
- размер поля подшивки для переплета;
- расстояние от текста до колонтитула;
- вид ориентации страницы (книжную или альбомную)
- способ выравнивания текстового поля на странице (по центру, по верху, по низу, по верху и низу одновременно);
- различающиеся колонтитулы на четных и нечетных страницах;
- особый формат первой страницы.

Для задания параметров формата нужно последовательно выбрать пункты:

#Макет# – !Параметры страницы!-[Открыть диалоговое окно]

Установить параметры страницы: поля, колонтитулы и др. сделать необходимые установки в окнах на вкладках всплывающей панели «Параметры страницы» (Рис. 1.2А, Б).

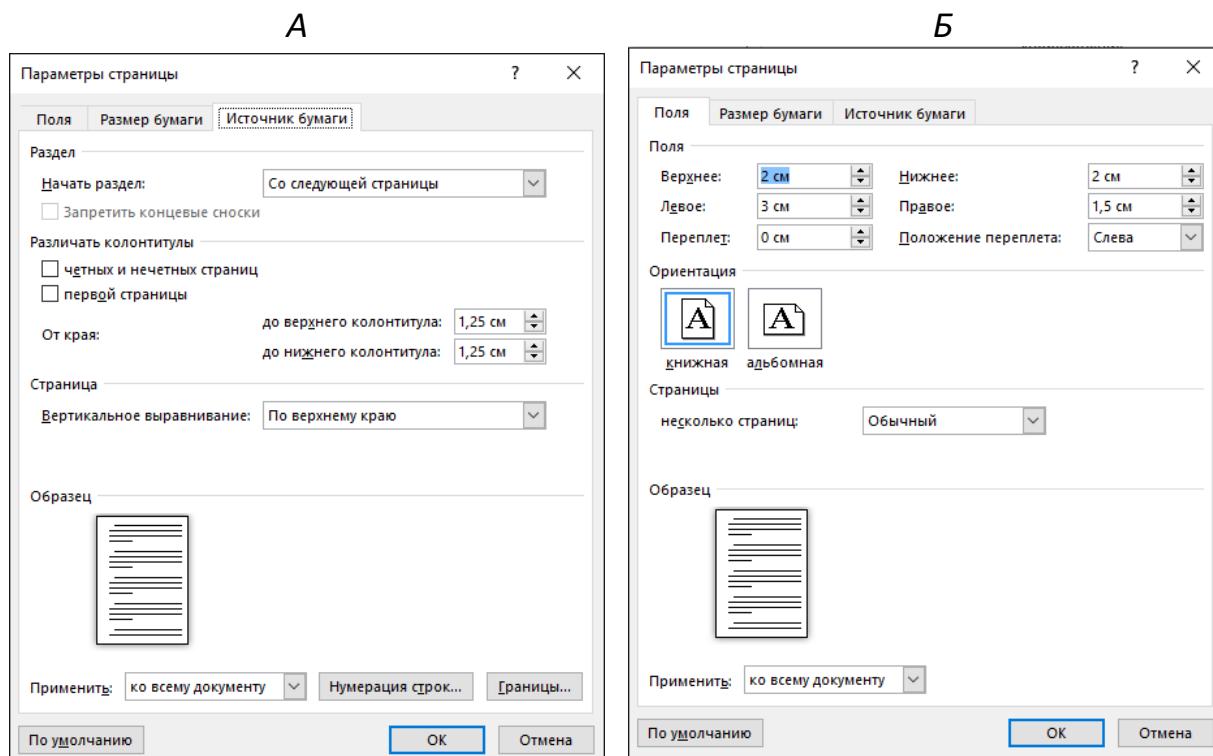


Рис. 1.1.

Действия по форматированию страницы можно выполнить также путем активации соответствующих пиктограмм на вкладке «Макет».

## Форматирование абзаца

Для задания параметров формата нужно последовательно выбрать пункты:

#Главная# – !Абзац! – [Открыть диалоговое окно];

При форматировании абзаца во всплывающей панели (Рис.1.3) устанавливаются левый и правый отступы, размер отступа в начале абзаца, размер межстрочного интервала и др.

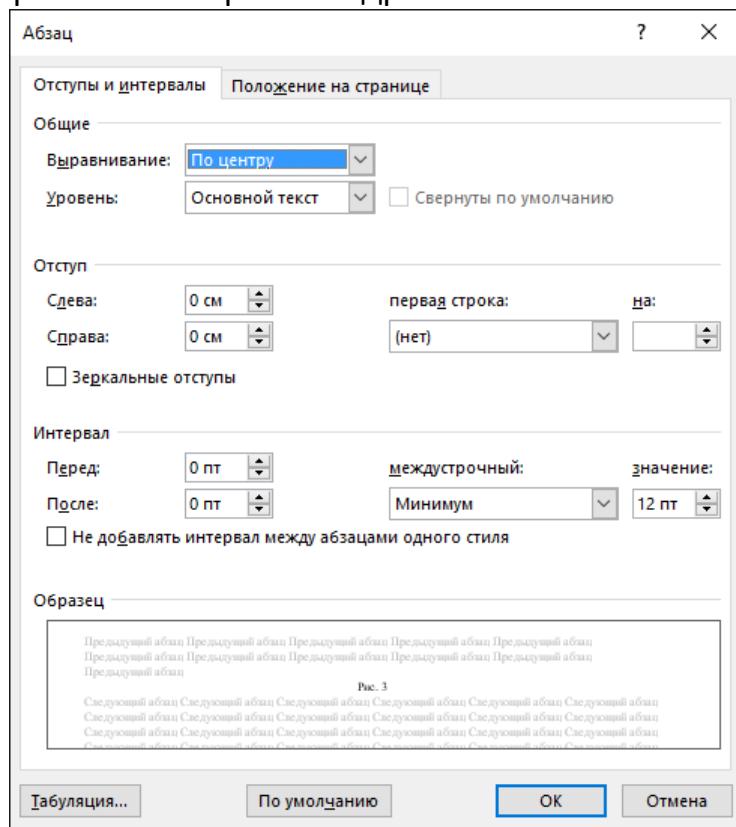


Рис. 1.2.

## Форматирование текста

При форматировании текста устанавливается вид шрифта (стиль написания), размер символов, интервал между символами и текстовые эффекты. Для перехода в режим форматирования текста нужно последовательно выбрать пункты:

#Главная# - !Шрифт!- [Открыть диалоговое окно];  
и далее в появившейся диалоговой панели (Рис. 1.4) в окнах «Шрифт», «Начертание» и «Размер», используя кнопки прокрутки, выбрать и задать необходимые установки щелчком мыши на соответствующих табличках, например: Шрифт - «Times New Roman», Начертание - «Курсив», Размер - "12". Переключение с одного алфавита на другой - клавишей <Ctrl+Shift><sup>4</sup> или <Alt+Shift>.

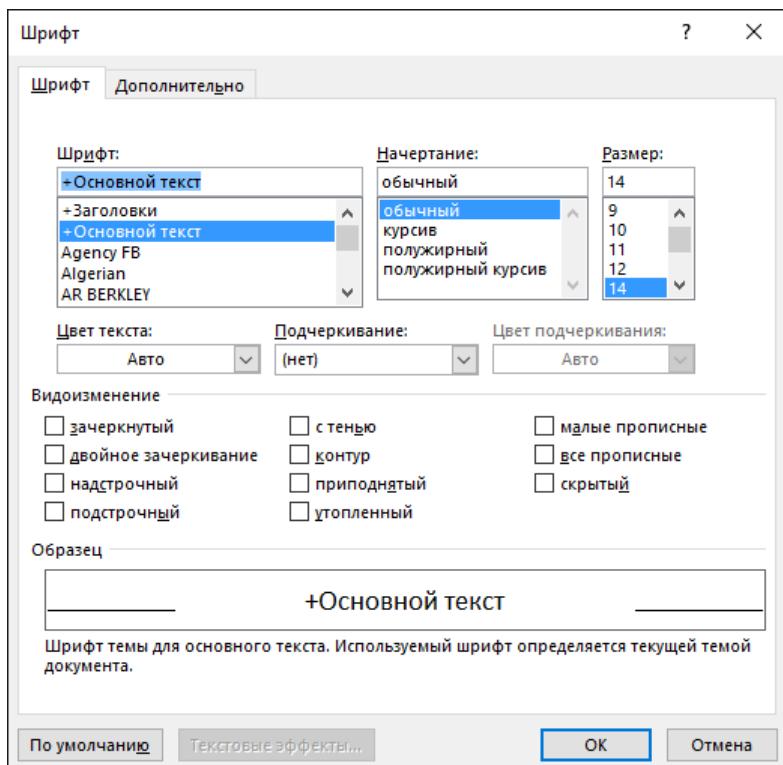


Рис. 1.3.

Можно задать “мягкий перенос”, т. е. с разбиением слов на слоги:  
#Макет# - !Параметры страницы! – [Расстановка переносов] - [Авто].

## Использование автотекста

Часто встречающийся фрагмент текста (например, фразу) можно сделать элементом автотекста, чтобы не повторять многократно его ввод с клавиатуры. Вместо ввода текста фрагмента достаточно будет ввести его имя и нажать клавишу <F3>.

Для создания элемента автотекста нужно:

- Пометить фрагмент текста;

---

<sup>4</sup> Здесь и в дальнейшем в треугольных скобках будут приводиться названия клавиш клавиатуры.

- b) <Alt+F3>;
- c) В диалоговом окне «Создание нового стандартного блока» ввести имя (коллекцию, категорию, описание) элемента автотекста.

Для ввода автотекста нужно:

- a) Установить курсор в место ввода;
- b) Ввести имя элемента автотекста;
- c) <F3>.

*Примечание: имя автотекста должно быть отделено пробелами от текста с двух сторон.*

Можно использовать стандартные элементы автотекста из коллекции Word 2016. Для этого:

- a) #Вставка# – !Текст! – [Просмотреть экспресс-блоки] – [Организатор стандартных блоков];
- b) Пометить стандартный блок из группы АВТОТЕКСТ- [Вставить];

## Вставка формул

### Вставка простых формул

- a) #Вставка# – !Символы! – [Уравнение] – [Вставить новое уравнение];
- b) Создать формулу, пользуясь элементами всплывающей вкладки Конструктор (Рис. 1.5).

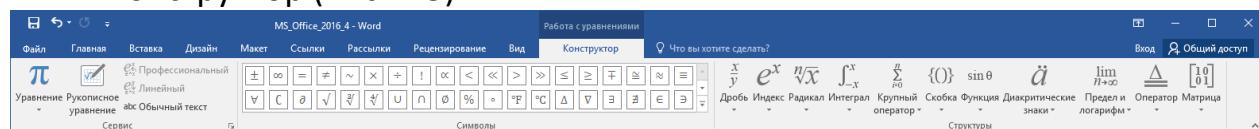


Рис.1.5

Можно сохранить созданную формулу в коллекции и присвоить ей имя:

- a) Вызвать контекстное меню;
- b) Во всплывающем диалоговом окне «Создание стандартного блока» задать имя (коллекцию, категорию, описание) блока

Для вставки формулы из коллекции:

- a) Установить курсор в место вставки;
- b) #Вставка# – !Текст! – [Просмотреть экспресс-блоки] – [Организатор стандартных блоков];
- c) Пометить стандартный блок из группы УРАВНЕНИЯ- [Вставить];

## Вставка иллюстраций

В документ могут быть вставлены:

- Растровый рисунок из файла формата bmp, jpg, gif;
- Копия графического экрана Windows;
- Созданный с помощью инструментов Word векторный рисунок.

Для вставки рисунка из файла нужно:

- a) Установить курсор в место вставки.
- b) #Вставка# – !Иллюстрации! - [Рисунки] - [Из файла...].

После вставки рисунка автоматически открывается вкладка «Формат» главного меню (Рис. 1.6), пользуясь пиктограммами которой можно изменить параметры рисунка: яркость, контрастность, размер и т. д.

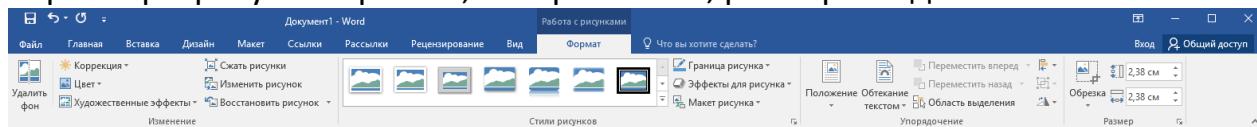


Рис. 1.6.

Для редактирования формата ранее вставленного рисунка нужно:

- a) Пометить рисунок;
- b) Открыть вкладку «Формат» двойным щелчком левой кнопки мыши на рисунке;
- c) Отредактировать параметры формата рисунка.

При редактировании настраивается:

- размер рисунка;
- способ обтекания рисунка текстом;
- вид выравнивания (влево, вправо, по центру);
- яркость и контрастность рисунка.

При выборе способа обтекания «за текстом» рисунок будет наложен на текст. Способ «за текстом» может быть использован, например, при создании документа с подложкой (с «водяными знаками»). Для этого, используя имеющиеся регулировки яркости и контрастности, цвет графического изображения или текста, которые представляют подложку, нужно взять существенно менее насыщенным, чем цвет основного текста.

Можно создать документ с текстовой или графической подложкой другим способом:

- a) #Дизайн# - !Фон страницы! – [Положка] – [Настраиваемая подложка];
- b) Выбрать рисунок или создать текст для подложки.

## Вставка таблиц

### *Создание простой таблицы*

Для создания простой таблицы:

- #Вставка# - !Таблицы! - [Таблица] – [Вставить таблицу];
- Задать количество колонок и строк, способ автоподбора ширины столбцов.

## Вставка фигур

Word 2016 предоставляет большую коллекцию фигур: прямоугольников, фигурных стрелок и др. Для вставки фигуры:

- #Вставка# - !Иллюстрации! – [Фигуры];
- Выбрать фигуру из всплывающей коллекции;
- Буксировкой курсора создать фигуру в документе и задать стиль фигуры, используя инструменты : Заливка фигуры, Контура фигуры, Эффекты фигуры на вкладке «ФОРМАТ» (Рис. 1.7) в группе «Стили фигур»;

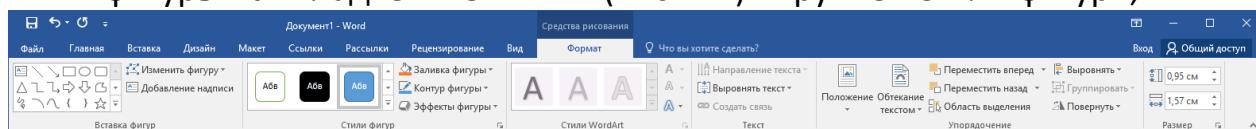


Рис. 1.7.

- Назначить способ обтекания фигуры текстом:

#Формат# - !Упорядочение! – [Обтекание текстом] – {Выбрать способ}. Можно задать позицию фигуры на странице: #Формат# - !Упорядочение! – [Положение] – {Выбрать способ}.

Двойным щелчком левой кнопки мыши на фигуре, ранее созданной в документе, открывается вкладка «ФОРМАТ», пользуясь пиктограммами которой можно изменять стиль фигуры, эффекты объема, способ обтекания текстом и др.

Можно создавать композицию из нескольких фигур. Для этого:

- #Вставка# - !Иллюстрации! – [Фигуры] – [Новое полотно];
- Создать необходимую композицию из фигур в поле полотна. При создании любой фигуры автоматически всплывает вкладка «ФОРМАТ» (рис. 1.7), в которой содержатся инструменты для работы с объектами на полотне;

- c) Для точной подгонки места расположения фигур использовать клавиши стрелок клавиатуры при нажатой клавише <Ctrl>;
- d) Для того, чтобы линии и стрелки между фигурами проходили строго горизонтально или вертикально, проводите их при нажатой клавише «Shift»;
- e) Для создания надписей на фигурах активировать вкладку «Средства рисования/Формат», вызвать «Добавление надписи», затем ввести и отформатировать текст, пользуясь инструментами «Заливка фигуры», «Контур фигуры», «Выровнять текст» и др. на вкладке «Формат». Эти же действия можно выполнить, пользуясь контекстным меню. Выравнивание текста по горизонтали выполняется инструментами на вкладке «Главная»
- f) Надписи соединительных линий создавать с помощью элемента «Надпись». Для вставки элемента «Надпись»:  
#Вставка# - !Иллюстрации! – [Фигуры] – [Надпись]  
Для точной подгонки места расположения элемента «Надпись» использовать клавиши стрелок клавиатуры при нажатой клавише <Ctrl>. Для того, чтобы контуры рамки элемента «Надпись» были невидимыми активировать вкладку «Средства рисования/Формат», далее:  
[Стили фигур] – [Контур фигуры] – [Нет цвета]  
Для того, чтобы контуры элемента «Надпись» не закрывали линии чертежа, нужно поместить элемент «Надпись» на задний план: [На задний план] из контекстного меню.
- g) Подогнать размер полотна к размеру созданного векторного рисунка. Для этого из контекстного меню полотна выбрать [Подогнать размер], затем [Увеличить];
- h) Отрегулировать масштаб полотна с векторным рисунком. Для этого выбрать [Изменение масштаба рисунка] из контекстного меню полотна, затем выполнить регулировку масштаба полотна с помощью мыши;
- i) Выполнить группировку элементов векторного рисунка в единый объект. Для этого обвести созданный рисунок контуром при нажатой левой кнопки мыши, затем [Группировка] – [Группировать];
- j) Задать способ обтекания полотна с рисунком текстом из контекстного меню полотна, например, [Форматировать полотно] – [Вокруг рамки], разместить полотно с рисунком в нужном месте создаваемого текстового документа и вставить название рисунка: [Вставить название].

## Проверка правописания

Встроенная проверка правописания позволяет обнаруживать и в диалоговом режиме исправлять грамматические и орфографические ошибки в тексте.

Для проверки орфографии и грамматики нужно:

- a) Пометить проверяемый фрагмент текста;
- b) #Рецензирование# - !Правописание! – [Правописание]

Слова, содержащие орфографические ошибки, и предложения, содержащие грамматические ошибки, будут помечены. При обнаружении ошибок возможные варианты правильного написания слова и исправления грамматических ошибок будут показаны на всплывающих панелях «Орфография» и «Грамматика».

Установить/отменить отдельные действия проверки орфографии и грамматики можно на панели «Параметры Word». Для доступа к этой панели:

#Файл# - [Параметры] – [Правописание]

## 1.3. Редактирование текста

При редактировании можно:

- дополнить текстовый файл;
- произвести необходимые изменения текста, удаляя, копируя и перемещая отдельные фрагменты текста;
- изменить шрифт, стиль написания и размер символов в любом фрагменте текста;
- скопировать формат выделенного фрагмента текста;
- изменить поля, прозвести выравнивание, задать отступы;
- ввести буквицу (первый символ абзаца).

Операции редактирования (удаление, копирование, перемещение фрагментов текста (слова, строки, предложения, строчного или прямоугольного блока текста) производятся после того, как данный фрагмент текста помечен.

## Пометка текста

Можно пометить:

- отдельное слово, сделав двойной щелчок мышью на нем;

- строку, найдя курсор маркировки (он появится при установке указателя выбора объекта (основной формы курсора мыши) в левом поле экрана перед строкой) и щелкнуть левой кнопкой мыши;
- абзац, если найти курсор маркировки в центре маркируемого абзаца и трижды щелкнуть левой кнопкой мыши;
- весь текст, если найти курсор маркировки в любом месте левого поля текста, нажать <Ctrl> и щелкнуть левой кнопкой мыши. Второй способ - #Главная# - !Редактирование! - [Выделить] - [Выделить все];
- одно предложение, если установить курсор редактирования в начало предложения, нажать <Ctrl> и щелкнуть левой кнопкой мыши;
- произвольный прямоугольный фрагмент текста, если установить курсор редактирования в левый верхний угол фрагмента, нажать <Alt> и левую кнопку мыши и, удерживая их нажатыми, "протащить" курсор мыши в правый нижний угол фрагмента, затем отпустить;
- фрагмент текста от текущей позиции текстового курсора до текстового курсора мыши. Для этого зафиксировать текстовый курсор в начале фрагмента текста, поместить тестовый курсор в конец фрагмента (но не фиксировать его!), затем нажать <Shift> и щелкнуть левой кнопкой мыши.

### **Вставка, копирование, перемещение фрагмента текста**

- a) пометить фрагмент текста;
- b) занести помеченный фрагмент в буфер хранения:  
#Главная# - !Буфер обмена! - [Копировать] - если нужно копировать или  
#Главная# - !Буфер обмена! - [Вырезать] - если нужно перенести;
- c) переместить курсор редактирования в место копирования или вставки;
- d) вставить или скопировать фрагмент:  
#Главная# - !Буфер обмена! - [Вставить].

С помощью буфера хранения можно переносить текстовые фрагменты не только в пределах одного документа, но также и из других источников: из другого документа Word, из окна Internet Explorer, из окна электронной почты, из окон документов других текстовых редакторов, из окон программ – приложений Windows.

Можно использовать «горячие» клавиши:

- <Ctrl+C> - копировать;
- <Ctrl+X> - вырезать;
- <Ctrl+V> - вставить.

## Изменение шрифта, стиля написания, размера символов

Для этого необходимо:

- a) пометить фрагмент текста;
- b) вызвать контекстное меню щелчком правой кнопки мыши;
- c) выбрать пункт «Шрифт»;
- d) произвести установки, описанные в разделе «Форматирование текста».

Некоторые изменения (введение жирного шрифта, курсива, подчеркивания) можно сделать, используя соответствующие пиктограммы, в автоматически всплывающей после пометки фрагмента текста панели инструментов для редактирования шрифта.

Для введения буквицы<sup>5</sup>:

- a) промаркировать абзац;
- b) #Вставка# - !Текст! - [Добавить буквицу] – [Параметры буквицы];
- в) во всплывающем диалоговом окне задать параметры буквицы (позицию (в тексте, на поле) и размер).

## Копирование формата

- a) пометить фрагмент текста, формат которого необходимо скопировать;
- b) #Главная# - !Буфер обмена! – [Формат по образцу];
- c) пометить курсором копирования формата (он будет иметь вид кисти) фрагмент текста, формат которого нужно установить по образцу.

*Примечание. Если сделать двойной щелчок левой кнопкой мыши на кнопке «Формат по образцу» на вкладке «Главная», то операцию копирования формата можно выполнять многократно. После окончания операций копирования формата нужно сделать однократный щелчок левой кнопкой мыши на пиктограмме «Формат по образцу»*

## Отмена последней операции редактирования

Сделать щелчок левой кнопкой мыши на пиктограмме «Отменить ввод», находящейся на панели быстрого доступа в верхней части окна Word над лентой.

---

<sup>5</sup> Первая буква первого слова абзаца, выделенная другим размером.

*Примечание: повторяя отмену можно отказаться от нескольких (до 100) ранее выполненных операций редактирования.*

### **Загрузка текстового файла в окно редактирования**

Текстовый файл, созданный в редакторе Word или в другом текстовом редакторе, например, WordPad, можно загрузить в окно редактирования. Для этого:

#Файл# - [Открыть] – [Этот компьютер]

Далее в появившемся диалоговом окне «Открытие документа» нужно выбрать диск, папку и имя загружаемого Вами файла. Окончание установок - щелчок мышью на клавише «Открыть».

Можно последовательно загрузить два или более файлов. Для того, чтобы увидеть их одновременно на экране дисплея

#Вид# - !Окно! - [Упорядочить все]

#### **1.4. Обработка сложного документа**

Обработка сложного документа, которым может быть пояснительная записка к курсовому или дипломному проекту, оригинал – макет книги, включает действия по созданию и форматированию титульной страницы, нумерации страниц, структурирование текста на главы, разделы и подразделы, формирование оглавления, предметного указателя, списков рисунков и таблиц.

#### **Создание титульной страницы**

Сложный документ обычно содержит титульную страницу, содержащую название документа и отличающуюся по оформлению от других страниц документа. В верхнем колонтитуле титульной страницы помещается название организации, в которой выполнена работа, или фамилии авторов, в нижнем – название издательства и год создания документа. Номер страницы на титульной странице не ставится.

Создать и оформить титульную страницу можно с использованием шаблонов оформления:

#Вставка# - !Страницы! – [Титульная страница]

## Структурирование документа. Использование шаблонов и стилей

Сложный многостраничный документ должен содержать несколько уровней структуры: главы, разделы, подразделы и т. д. Рекомендованное количество уровней структуры – 3.

Для ускорения создания хорошо оформленного документа в сети Интернет на сайте [templates.office.com](http://templates.office.com) имеются шаблоны типовых документов: деловых писем, факсов, резюме и т. д.

Для поиска шаблона: #Файл# - [Создать] – [Поиск шаблона в сети], далее выбрать категорию шаблона, например, Бизнес, Личные, Печать и др. (см. рис. 1.8), выбрать в этой категории подходящий шаблон и нажать кнопку «Создать».

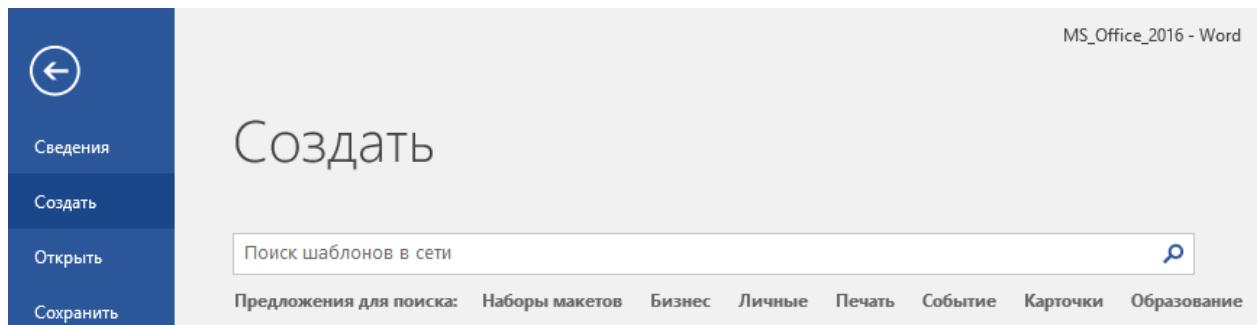


Рис. 1.8.

Можно создать собственный шаблон документа. Для этого можно отредактировать стандартный шаблон и сохранить его, указав тип файла «Шаблон Word» в папке «Документы/Настраиваемые шаблоны Office». Для того, чтобы использовать этот шаблон при создании следующих документов нужно вместо категории шаблонов «Рекомендуемые» выбрать категорию «Личные» (рис. 1.9):

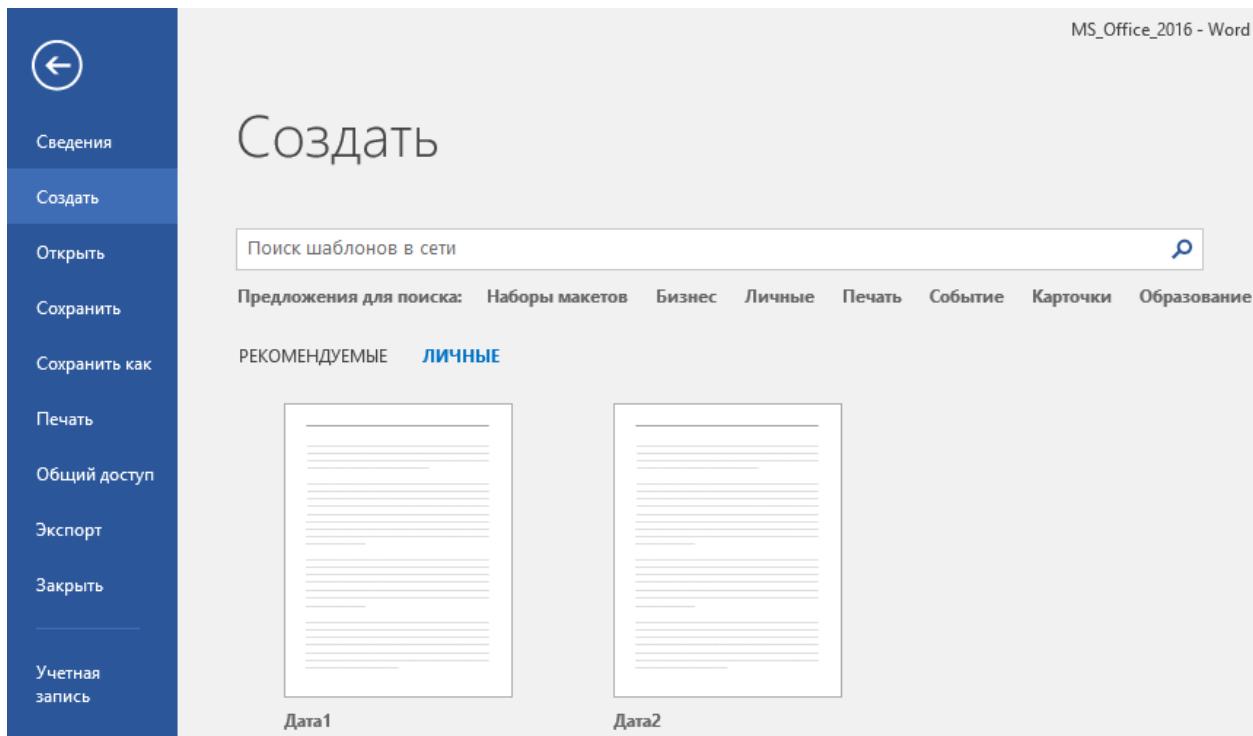


Рис. 1.9.

Для оформления заголовков глав, разделов и подразделов документа в шаблоне Normal.dot содержатся так называемые стили заголовков. Их использование дает возможность автоматизировать процесс создания оглавления и редактирования структуры документа.

Для присвоения стиля заголовку или другому фрагменту документа нужно:

- пометить заголовок;
- выбрать соответствующую пиктограмму, например, «Заголовок 2» в группе «Стили»;

Если необходимый стиль отсутствует, нужно отобразить все возможные стили в виде списка:

- #Главная# - !Стили! – [Открытие окна стилей] – [Параметры] – [Отображаемые стили] – [Все];
- выбрать стиль.

Формат любого стиля можно изменить. Для этого нужно последовательно выполнить следующие действия:

- выделить фрагмент текста, например, заголовок, стиль которого нужно изменить;
- #Главная# - !Стили! – [Открытие окна стилей] – {Активировать кнопку «Управление стилями»};

- c) Во всплывающем окне «Управление стилями» (рис. 1.10) выбрать из списка редактируемый стиль;

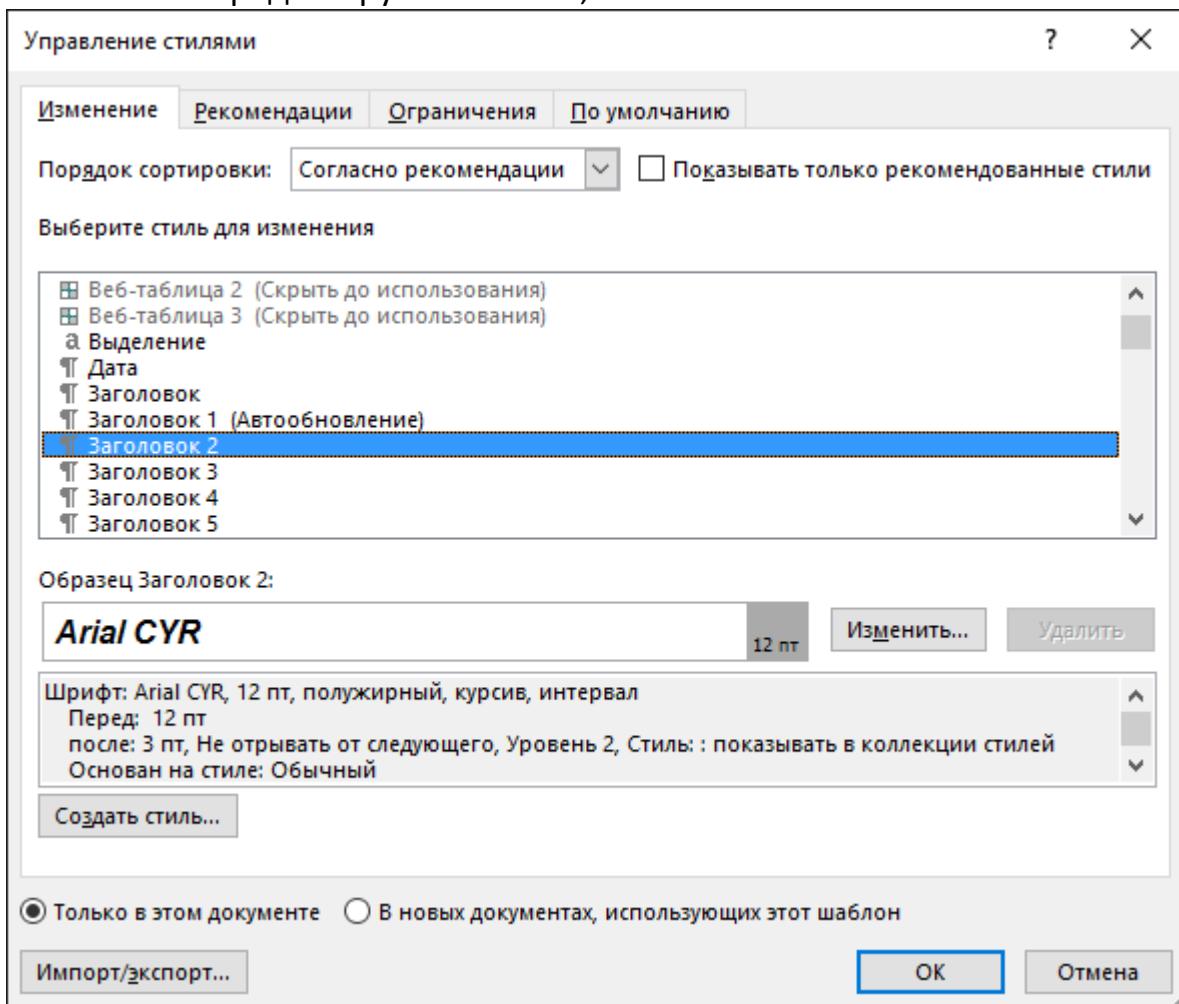


Рис. 1.10.

- a) в окне с названием стиля открыть меню, выбрать пункт «Изменить...» и во всплывающей панели «Изменение стиля» (рис.1.11)сделать необходимые установки .

Можно создавать также новые стили. Для этого:

- #Главная# - !Стили! – [Открытие окна стилей] – [Создать стиль];
- во всплывающей панели «Создание стиля» (рис. 1.12) сделать необходимые установки.

# 1. Microsoft Word 2016

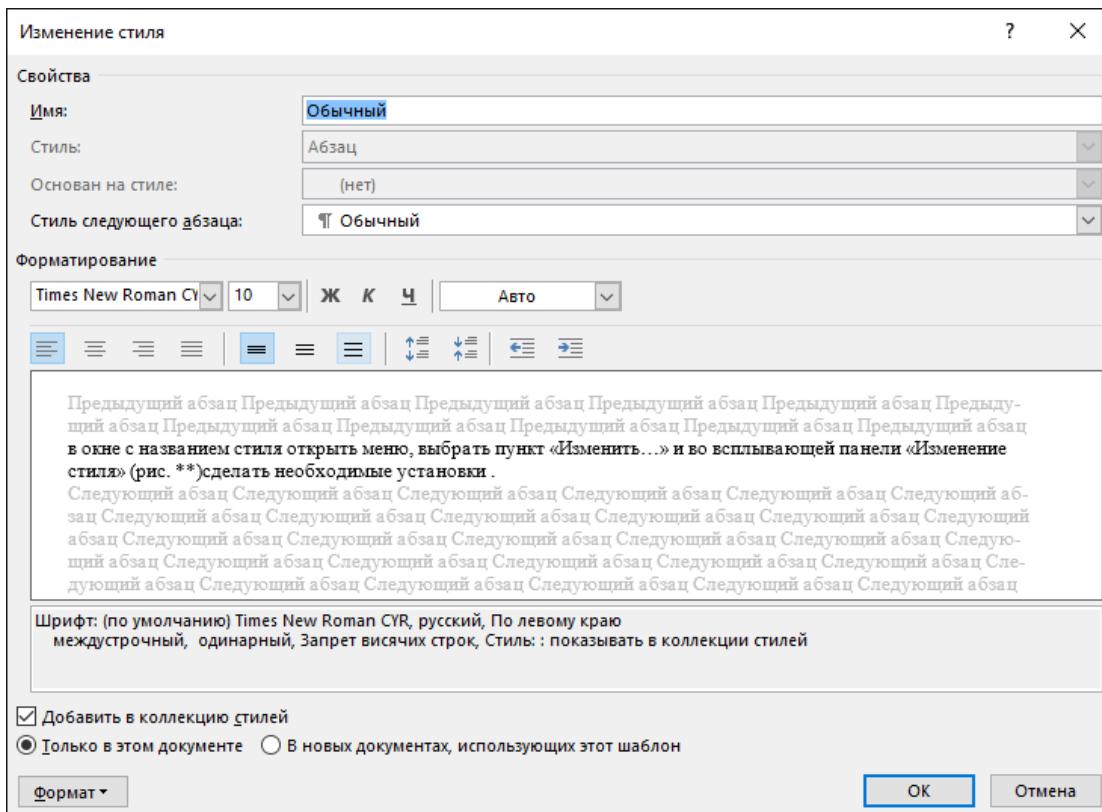


Рис. 1.11.

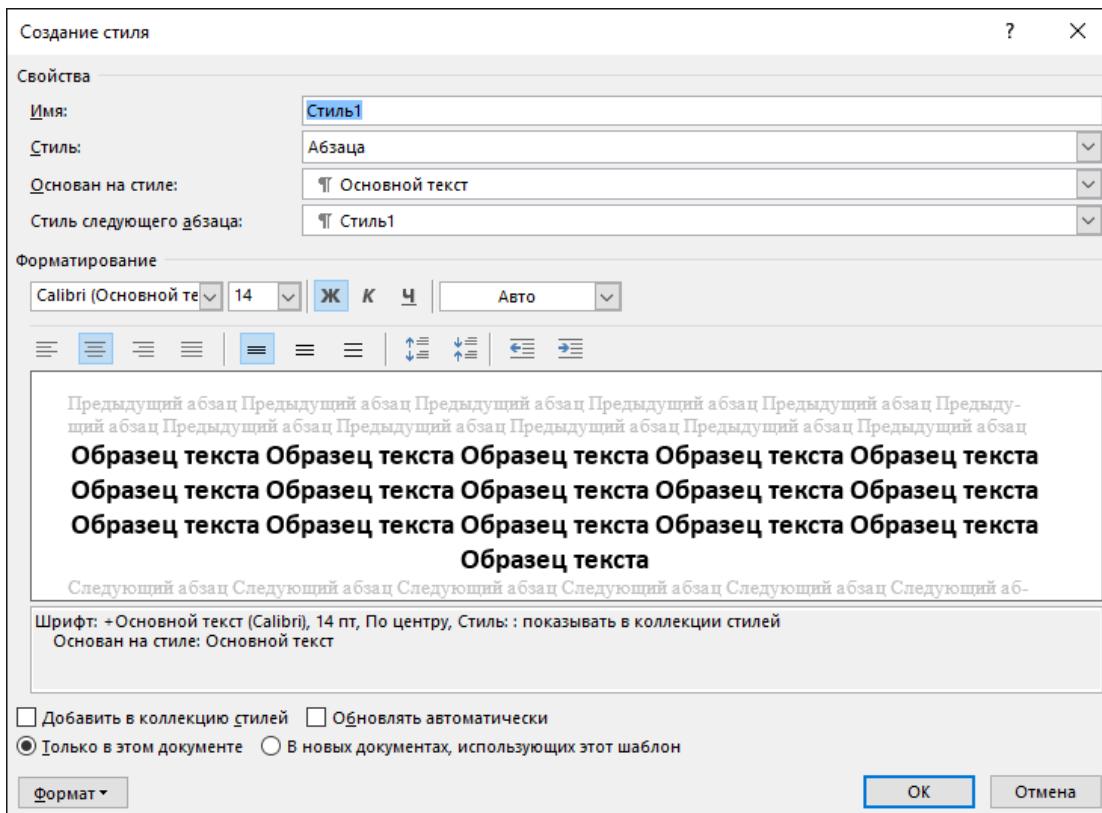


Рис. 1.12.

## Введение колонтитулов

- #Вставка# - !Колонтитулы! – [Верхний/Нижний колонтитул]
- выбрать из выпадающей галереи макет колонтитула или создать свой формат колонитула, выбрав пункт [Изменить верхний/нижний колонтитул];
- ввести текст в позиции, указанные в макете или в поле колонтитула.
- во всплывающей вкладке «Конструктор» (рис.1.13) установить параметры колонтитула, например: «Особый колонтитул для первой страницы», «Разные колонтитулы для четных и нечетных страниц»;

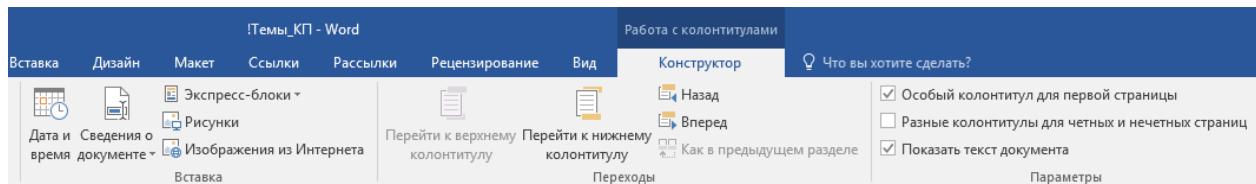


Рис. 1.13

Можно установить автоматический ввод в состав колонтитула ФИО автора, заголовок текущего раздела документа и другие элементы. Для ввода в колонтитул названий заголовков первого/второго уровня нужно последовательно выполнить следующие действия:

- #Вставка# - !Колонтитулы! – [Верхний/Нижний колонтитул] – [Изменить верхний/нижний колонтитул] – [Экспресс-блоки] - [Поле...];
- выбрать требуемое поле из списка, например, Author, Date, StyleRef для ввода в колонтитулы фамилии автора документа, даты или стиля заголовка на текущей странице соответственно;
- При выборе поля StyleRef далее выбрать нужное имя стиля из списка, например, Заголовок 1/Заголовок2 в окне ввода панели «Поле»:

## Нумерация страниц

- #Вставка# - !Колонтитулы! [Номер страницы...];
- выбрать из коллекции вариант расположения номера на странице;
- выбрать из коллекции вариант формата номера (задать положение и формат номера страницы, начальный номер). Для того, чтобы номер страницы не отображался на первой странице нужно во всплывающей вкладке «Конструктор» (рис.1.14) установить параметр «Особый колонтитул для первой страницы» и в окне «Формат номера страницы» установить «начать с 2».

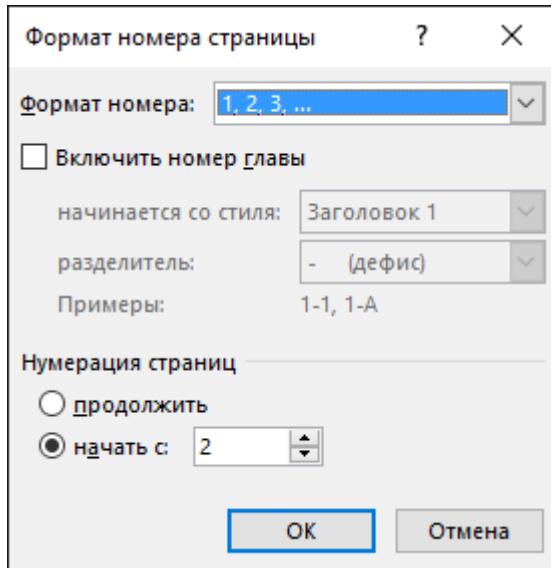


Рис 1.14.

## Формирование сносок

- поместить текстовый курсор в то место документа, где должен быть помещен маркер ссылки на сноска;
- #Ссылки# - !Сноски! - [Вставить сноска/Вставить концевую сноsku]. Обычная будет в конце страницы, концевая - в конце текста;
- ввести текст сноски.

*Примечание: посмотреть текст страницы со сноской можно только в режиме разметки страницы.*

## Маркировка и нумерация абзацев

- пометить абзацы<sup>6</sup>;
- #Главная# - !Абзац! – [Маркеры/Нумерация/Многоуровневый список] ;
- задать макет маркировки или нумерации.

Символом маркировки можно сделать рисунок. Для этого нужно выбрать стандартный рисунок в окне «Библиотека маркеров» (см. рис. 1.15.А) или, нажав кнопку «Определить новый маркер» в окне «Библиотека маркеров», выбрать какой-либо рисунок из файла (см. рис. 1.15.Б). После задания многоуровневой нумерации и выбора макета нумерации вначале создается одноуровневый список. Пользователь может понизить/повысить уровень любого абзаца, добавив/убрав в его начале символ табуляции клавишей <Tab>/<Shift+Tab>.

<sup>6</sup> Абзац – фрагмент текста, который заканчивается переводом строки.

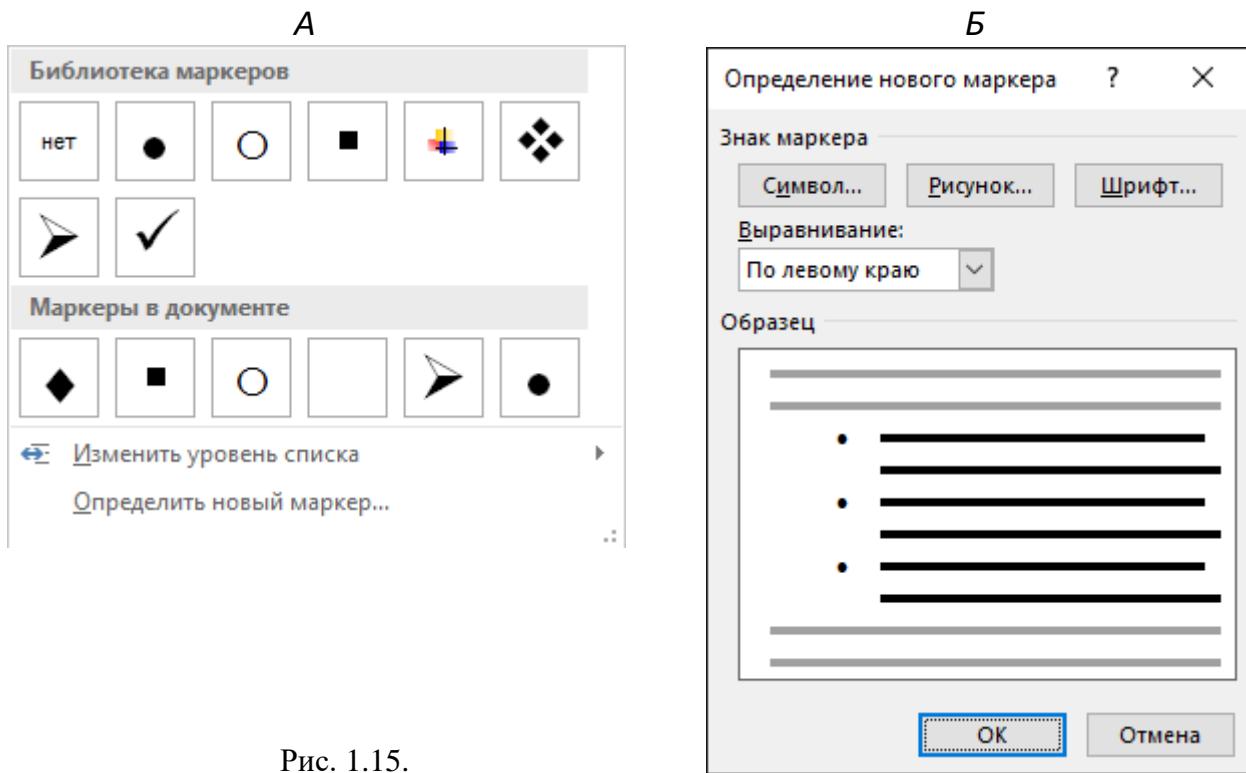


Рис. 1.15.

## Обрамление и заливка абзацев

- пометить абзац;
- #Главная# - !Абзац! - [Заливка]

и далее во всплывающей панели «Цвета темы» выбрать необходимый цвет заливки.

Можно задать обрамление и заливку абзаца/страницы:

- #Дизайн# - !Фон страницы! – [Границы страниц];
- во всплывающем диалоговом окне «Границы и заливка» рис. 1.16 установить необходимые опции.

## Многоколоночное расположение текста

- #Макет# - !Колонки! - [Другие столбцы];
- Установить во всплывающем окне «Колонки» рис. 1.17 количество колонок и другие параметры.

Для перехода от обычного "одноколоночного" заголовка к многоколоночному тексту на той же странице:

1. Microsoft Word 2016

#Макет# - !Параметры страницы! – [Разрывы] – [Разрывы разделов] - [Без разрыва].

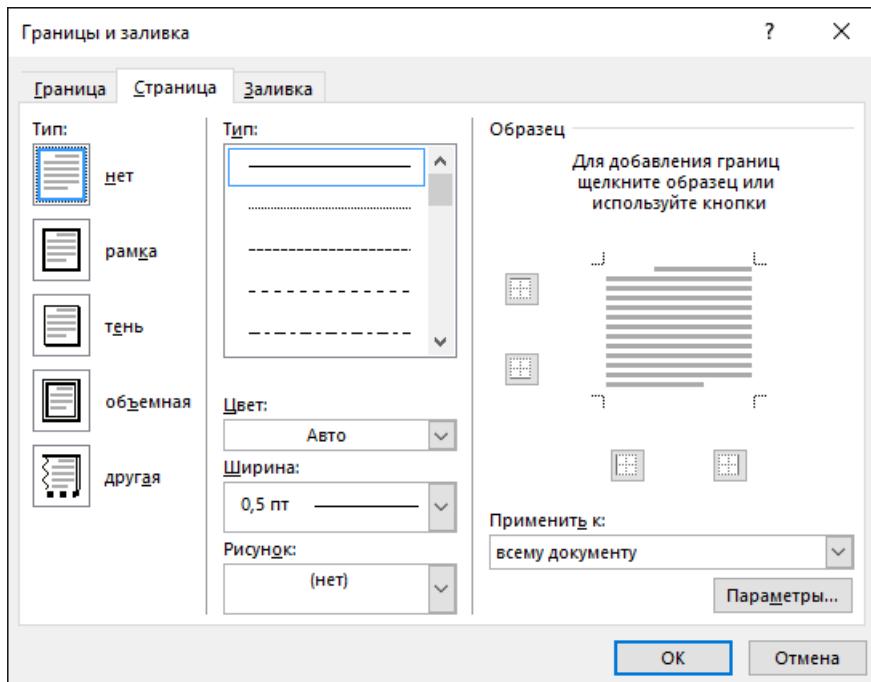


Рис. 1.16.

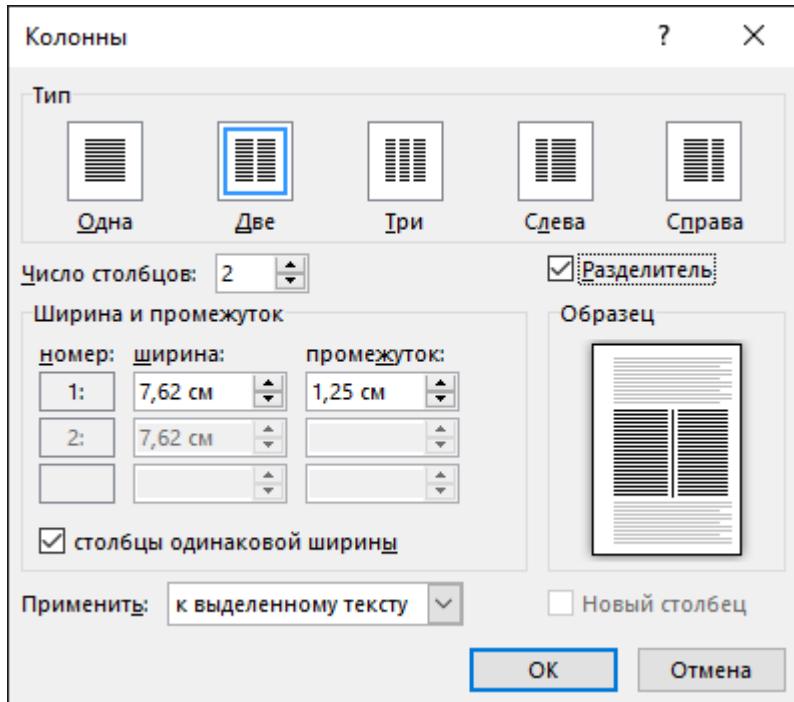


Рис 1.17

## Автонумерация рисунков

Для нумерации нужно пометить рисунок, затем вызвать контекстное меню и выбрать пункт «Вставить название».

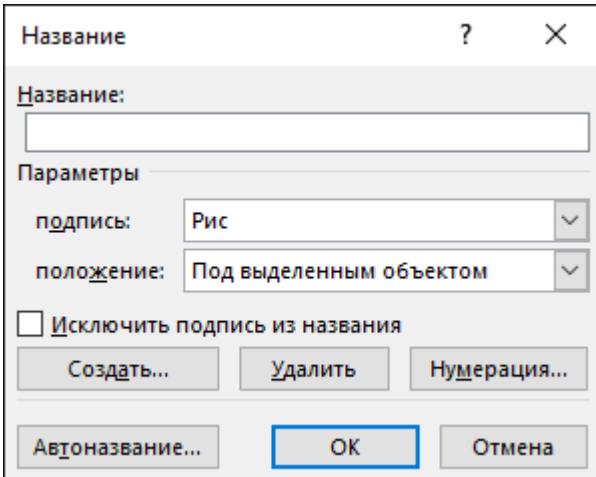


Рис. 1.18.

Можно воспользоваться предложенным форматом нумерации, изменить формат, создать собственный формат, щелкнув левой кнопкой мыши на кнопке «Нумерация» или «Создать», а также добавить к номерам рисунков подрисуночные подписи. При активизации кнопки «Автоназвание» во всплывающем диалоговом окне «Автоназвание» можно задать опции автонумерации, в результате чего нумерация иллюстраций (например, точечных рисунков) будет производиться автоматически.

При необходимости создания подрисуночной подписи к рисунку с обтеканием текстом нужно вначале назначить способ обтекания текстом для рисунка. Формат подрисуночной подписи можно изменить (см. раздел «Структурирование документа. Использование шаблонов и стилей»)

Для создания списка иллюстраций:

- поместить курсор в то место документа, в котором должен быть создан список иллюстраций;
- #Ссылки# - !Названия! – [Список иллюстраций].

При изменениях в составе иллюстраций для обновления списка иллюстраций:

- пометить список иллюстраций;
- вызвать контекстное меню и выбрать пункт «Обновить поле».

## Вставка сложных формул

Для вставки сложных формул нужно:

- Установить курсор в место вставки;
- #Вставка# - !Текст! – [Объект] – [Объект] ;
- Во всплывающем окне «Вставка объекта» выбрать “Microsoft Equation 3.0”;
- Используя всплывающую панель инструментов создать формулу;
- Сделать щелчок левой кнопкой мыши вне поля формулы.

Далее, используя средства создания специальных знаков на всплывающей панели инструментов Microsoft Equation (Рис. 1.19), создать формулу.

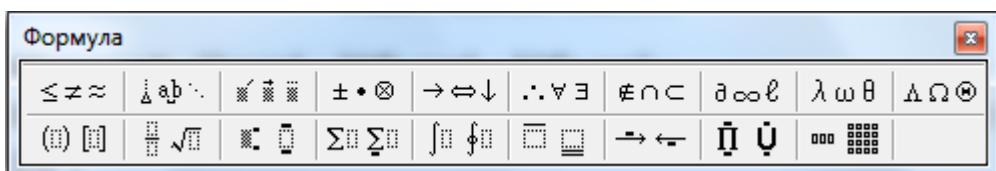


Рис. 1.19.

### **Создание сложной таблицы**

Сложная таблица получается из простой путем разбиения и объединения клеток. Для разбиения:

- Пометить клетку таблицы;
- #Макет# - !Объединение! – [Разбить ячейки].
- Задать количество столбцов и строк, на которые нужно разделить клетку.

Для объединения:

- Пометить клетки таблицы.
- #Макет# - !Объединение! – [Объединить ячейки].

На вкладке «Макет» также имеются другие средства управления (пиктограммы) для выполнения редактирования таблицы, такие как выравнивание текста в клетках таблицы, удаление и добавление столбцов и строк и т. д.

### **Построение и вставка графиков**

Для построения и вставки графиков:

- #Вставка# – !Иллюстрации! – [Диаграмма];

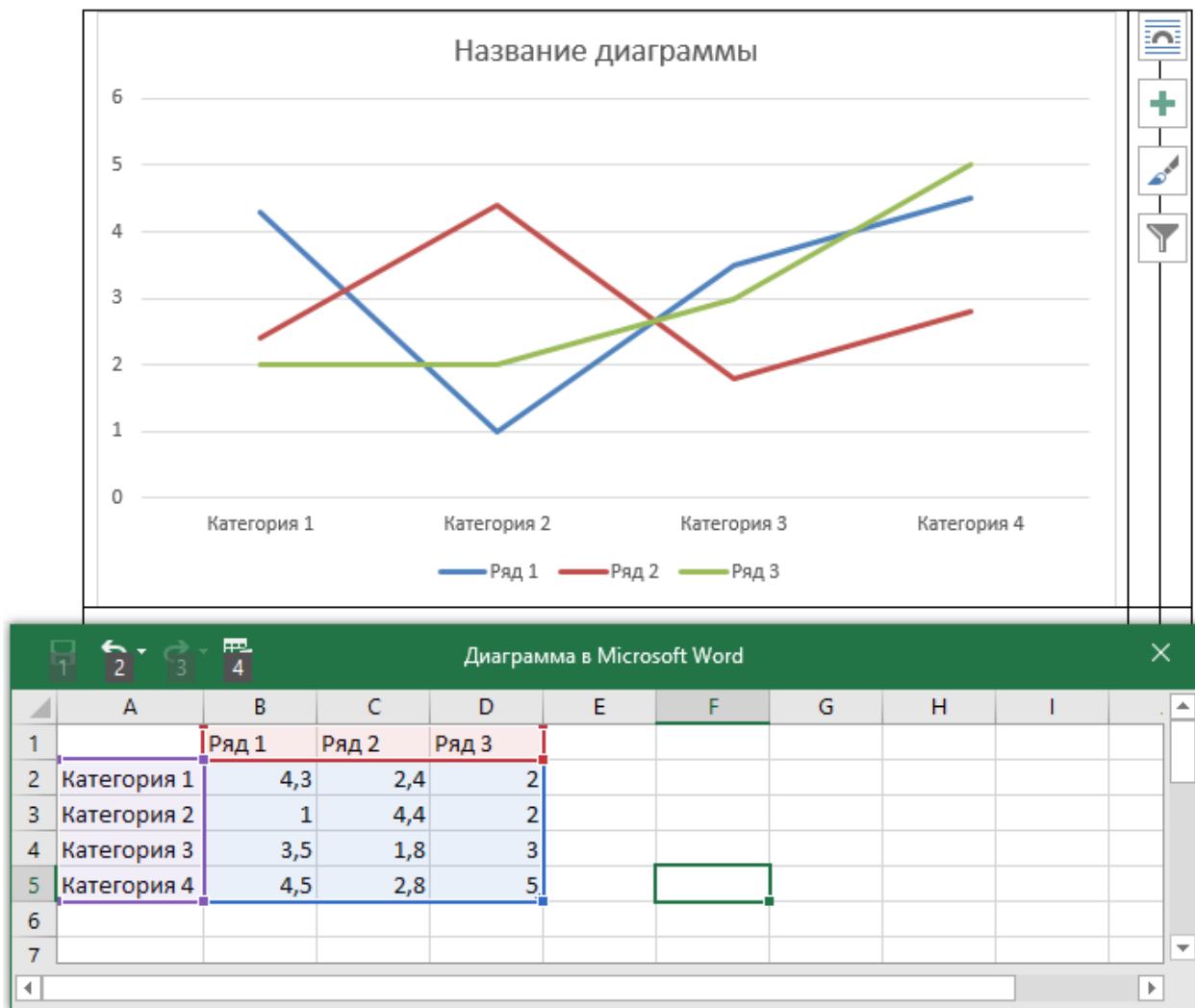


Рис. 1.20.

- b) заполнить данными для построения графика макет таблицы (см. рис. 1.20);
- c) отредактировать параметры формата графика (например, ввести заголовок), используя инструменты на вкладке «Формат» или вызвав соответствующий пункт в контекстном меню графика.

### Автоформирование оглавления

- a) выделить последовательно названия глав, разделов и подразделов документа и присваивать им стили Заголовок 1, Заголовок 2, Заголовок 3 соответственно;

- b) установить курсор в то место текста, где должно быть сформировано оглавление (например, в начале новой страницы после текста документа);
- c) #Ссылки# – !Оглавление! – [Настраиваемое оглавление...]
- d) выбрать в диалоговом окне рис. 1.21 формат оглавления и количество уровней заголовков

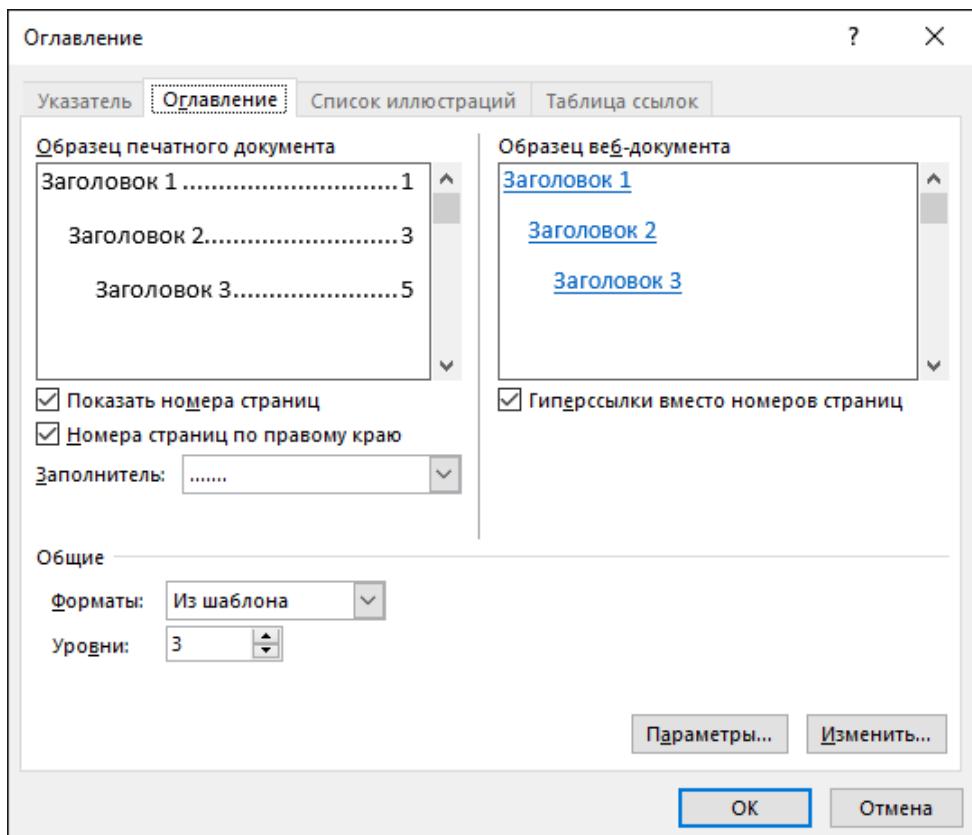


Рис 1.21.

Последовательность и уровни разделов и подразделов документа, отраженные в оглавлении, образуют так называемую структуру документа. Структуру документа можно изменить. Для этого перейти в режим «Структура»:

#Вид# - !Режимы! – [Структура]

и произвести перемещение или изменение уровней путем боксировки курсором мыши.

*Примечание. После внесения изменений в структуру документа (добавления новых разделов или перемещения разделов) нужно обновить оглавление, для чего вызвать контекстное меню из оглавления, выбрать в меню «Обновить поле».*

## **Автоформирование списка литературы**

- a) установить курсор в то место документа, где должна быть вставлена ссылка ;
- b) #Ссылки# - !Ссылки и списки литературы! – [Вставить ссылку] – [Добавить новый источник];
- c) заполнить поля сведений о книге или статье во всплывающем диалоговом окне «Создать источник».
- d) установить курсор в то место документа, где должен быть вставлен список литературы;
- e) #Ссылки# - !Ссылки и списки литературы! – [Список литературы].

При необходимости внесения изменений в формат ссылки:

#Ссылки# - !Ссылки и списки литературы! – [Управление источниками].

## **Создание списка иллюстраций**

- a) произвести нумерацию или автонумерацию рисунков таким образом, как описано в разделе «Автонумерация рисунков»;
- b) поместить курсор в место вставки списка иллюстраций ;
- c) #Ссылки# – !Названия! – [Список иллюстраций] .

## **Создание предметного указателя**

- a) пометить фрагмент текста, помещаемого в указатель;
- b) #Ссылки# – !Предметный указатель! - [Пометить элемент] - [Пометить все] – [Закрыть]
- c) установить курсор в то место текста (например, на последнюю страницу), где должны быть размещены указатели;
- d) #Ссылки# - !Предметный указатель! - [Предметный указатель] ;
- e) Установить параметры формата предметного указателя в окне «Указатель» рис. 1.22.

## 1. Microsoft Word 2016

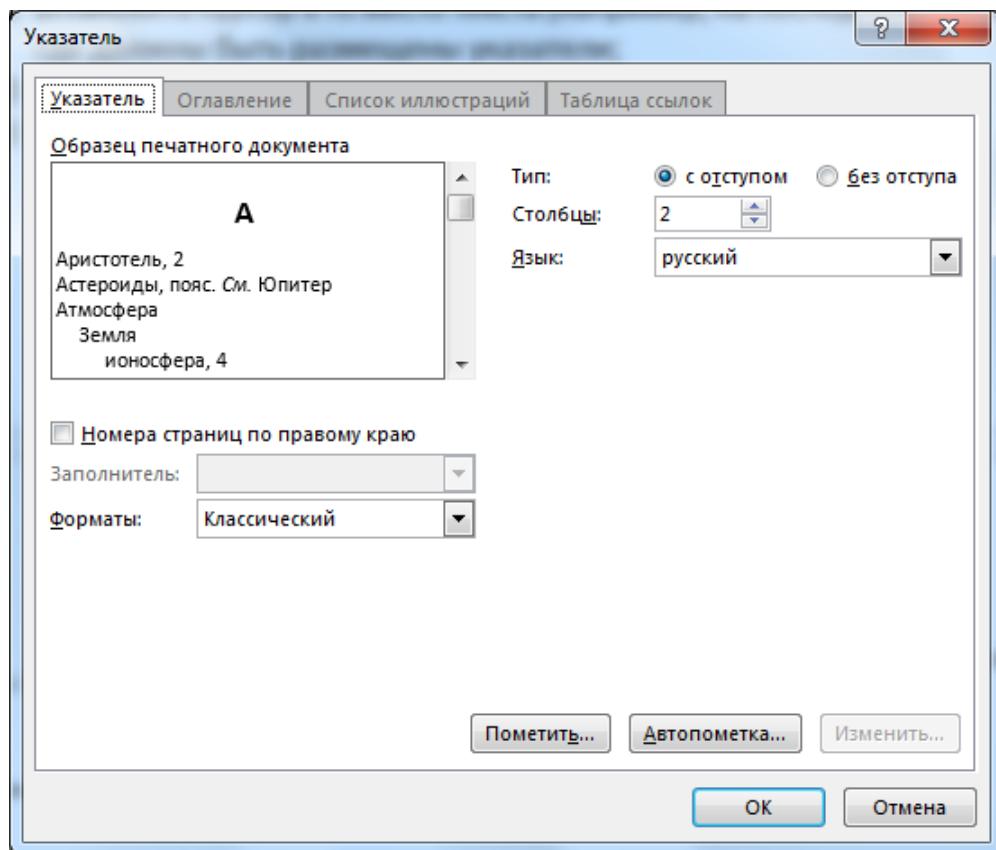


Рис. 1.22.

*Пример общего вида предметного указателя*

### Предметный указатель

<p><b>H</b></p> <p>HomePNA ..... 2, 6, 7</p>	<p><b>K</b></p> <p>Кольцо ..... 2, 11, 16, 28</p>
<p><b>W</b></p> <p>Wi-Fi ..... 7, 17, 25, 26, 27</p>	<p><b>П</b></p> <p>протокол ..... 6, 14, 19, 20, 22, 23, 26</p>
<p><b>З</b></p> <p>Звезда ..... 2, 10</p>	

*Импорт объектов из других Windows - приложений*

### Создание объекта и вставка

- установить курсор в позицию вставки объекта (рисунка Paint, таблицы Excel 2016, формулы Microsoft Equation и др.);
- #Вставка# - !Текст! – [Объект] – [Объект] ;

- c) Во всплывающем окне «Вставка объекта» выбрать “Microsoft Equation 3.0”;
- d) выбрать в появившемся меню нужное Windows- приложение (Excel 2016, Access, Paint и др.) в котором будет создаваться импортируемый объект. В результате внутри окна Word всплывает окно нужного Windows-приложения;
- e) создать объект в данном приложении;
- f) вернуться в окно Word, сделав щелчок левой кнопкой мыши вне поля созданного объекта.

### **Вставка текста из файла**

- a) установить курсор в позицию вставки объекта;
- b) #Вставка# - !Текст! – [Объект] – [Текст из файла] ;

### **Редактирование формата рисунков**

Производится после пометки рисунка, вызова контекстного меню и выбора пунктов «Формат рисунка...» и «Размер и положение...». Устанавливаются:

- размер рисунка;
- способ обтекания рисунка текстом;
- способ выравнивания.

Точную подгонку расположения рисунка на странице можно произвести, используя клавиши перемещения при нажатой клавише <Ctrl>.

Дополнительные возможности работы с рисунками (Художественные эффекты, коррекция яркости, контрастности, резкости и др.) представляются в пиктографическом меню на вкладке «Формат».

## **1.5. Создание прикладных документов**

### **Создание таблиц с вычисляемыми полями и полями со списком**

### **Создание вычисляемых полей в таблицах**

В клетки таблицы, создаваемой средствами Word 2016, можно вводить текст, цифровые данные и формулы. Формула может содержать числовые

значения, знаки математических операций, ссылки на клетки таблицы (например =D2\*E2)

Для создания вычисляемой клетки:

1. Установить курсор в клетку;
2. #Макет# - [Данные] - [Формула...];
3. В диалоговом окне “Формула” заменить формулу, предлагаемую по умолчанию на требуемую (см. рис. 1.23).

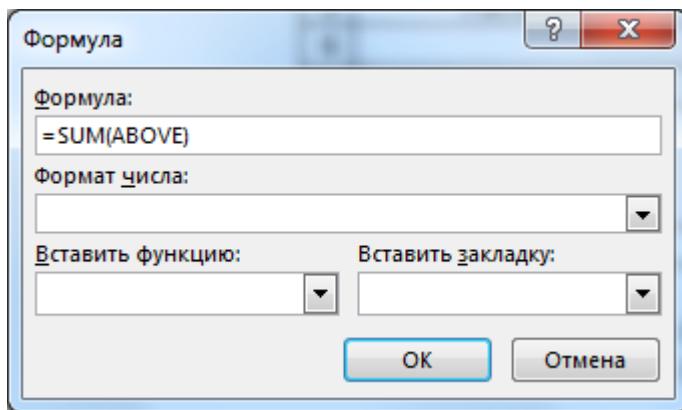


Рис. 1.23.

Формула должна начинаться со знака "=" и может содержать числа, номера клеток таблицы, знаки арифметических действий (+, -, \*, /, ^), %, операторы сравнения (=, <, >, >=, <=, <>).

Клетки нумеруются по горизонтали буквами (Внимание! В каждой строке порядок нумерации может быть свой, если произведено объединение ячеек!), а по вертикали - цифрами.

Сразу же после ввода формулы в клетке будет отображен результат. Чтобы посмотреть формулы в клетках таблицы, нужно пометить таблицу и нажать клавиши <Shift + F9>. Этой же комбинацией клавиш можно возвратиться к обычному изображению таблицы.

*Внимание! При изменении содержимого ячеек с цифровыми данными вычисления по формулам автоматически не производятся! Для обновления содержимого вычисляемых ячеек - нажать <F9>.*

Ниже приведен пример фрагмента платежного документа. Колонка «Сумма» в таблице Табл. 1.1 вычисляется по формулам.

Табл. 1.1

**Поставщик:** ЗАО "Стройинвест"

**Банк поставщика:** Адмиралтейский филиал АО ПСБ г.С.-Петербург, к/с 751911209, р/с 512000366676 МФО 161044, 4400300742 ИНН 7804004533  
ТОО "Стройимпульс"

**СЧЕТ №**

от .2015-10-08

№	Наименование	Ед. изм.	Кол-во	Цена	Сумма
1	Белила цинковые	кг	10	34,00	340
2	Кисти	шт	20	40,50	810
3	Растворитель	литр	10	18,00	180
4					
5					
6					
7					
8					
				ИТОГО	1330
				НДС (20%)	266
				ВСЕГО	1596

Сумма прописью:

Подписи:

### ***Создание полей со списком в таблицах***

Поля со списком в таблицах дают возможность создавать серию документов с помощью одного шаблона, например, табл. 1.2.

Табл.1.2

Поступ. в банк	ПЛАТЕЖНОЕ ПОРУЧЕНИЕ	101		0401002
		ДЕБЕТ		СУММА
Плательщик	ЗАО "Ритм" ИНН 8804046640	111	Сч. 3447176 700161244	
Банк плательщ.	Отд. Мостбизнесбанка в СПб	04430659		
Получатель			Сч. 003911103	
Банк получателя	ООО Альфа	181042	700151413	

Сумма прописью			пеня за дней из %руб сумма с пеней
Дата получ. товара, оказания услуг. Назнач. платежа, наимен. товара, выполненных работ, оказания услуг, №№ и суммы товарных документов			Вид обраб
			Вид опер
			Очер плат
М.П.	Подписи клиента	Проведено банком	Призн фз/пл
		Подписи банка	Назн пл
			Срок пл

Для создания формы поля (клетки):

- a) пометить клетку;
- b) #Разработчик<sup>7</sup> - !Элементы управления! – [Поле со списком] – [Свойства];
- c) во всплывающем окне задать последовательно названия пунктов поля со списком, (например, «ООО Альфа», см. табл. 1.2)

---

<sup>7</sup> Вкладку «Разработчик» нужно добавить на ленту: #Файл# – [Параметры] – [Настроить ленту] – [Основные вкладки/Разработчик]

# 1. Microsoft Word 2016

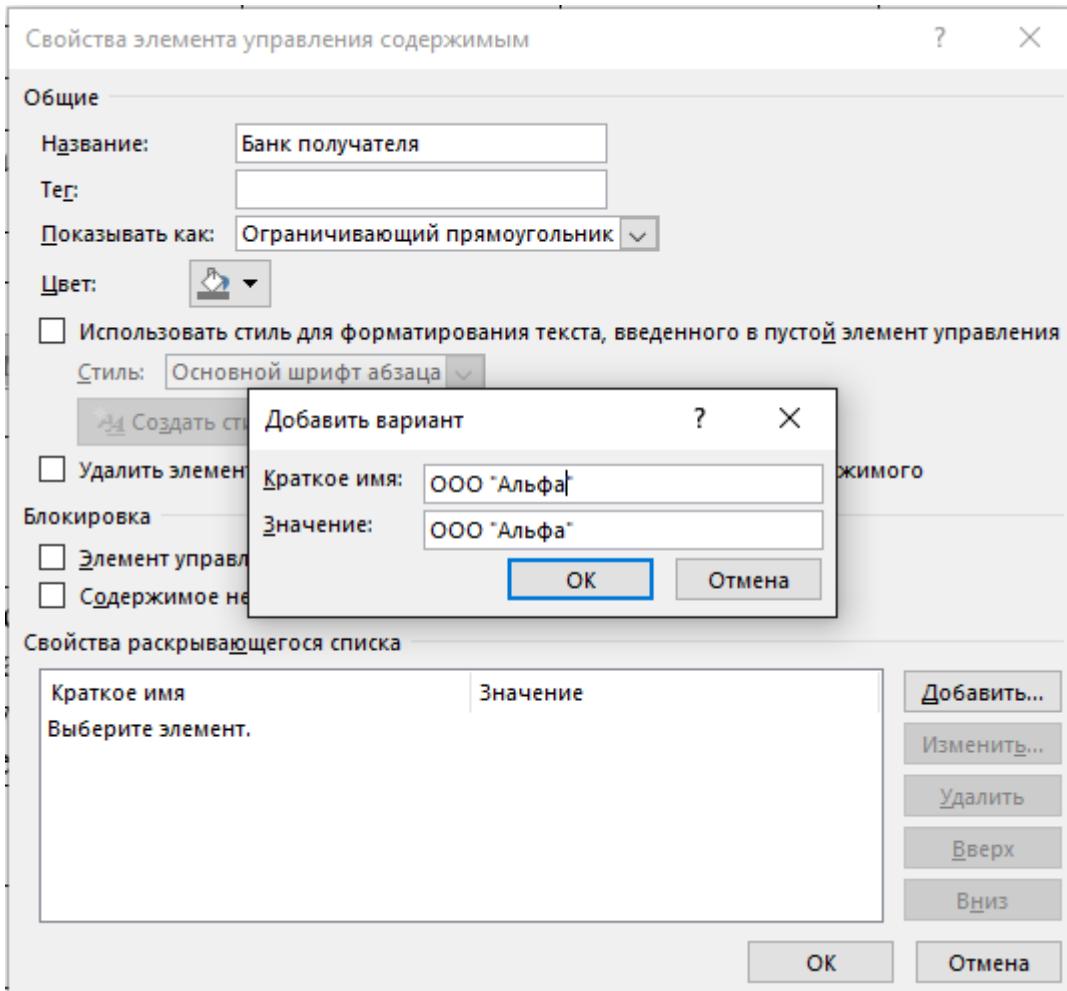


Рис.1.24.

Для ввода списка (например, названий или платежных реквизитов предприятий) в клетку, форма которой - поле со списком:

- [Добавить];
- ввести последовательно элементы списка в диалоговое окно «Добавить вариант» (Рис. 1.24). В результате будет получен список рис. 1.25.

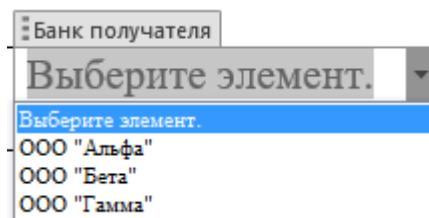


Рис. 1.25.

## Создание документов путем слияния

Данный прием создания документов предназначен для ускорения создания серий однотипных документов, различающихся только отдельными элементами (например, ФИО и другими реквизитами адресата). Для создания документов путем слияния нужно создать файл основного документа (например, письмо) и файл-источник данных (например, с реквизитами адресатов).

### *Создание таблицы-источника данных*

- #Рассылки# – !Начало слияния! – [Начать слияние] - [Письма];
- #Рассылки# – !Начало слияния! – [Выбор получателей] - [Ввести новый список] ;
- Во всплывающем диалоговом окне "Новый список адресов" выбрать «Настройка столбцов»;
- Во всплывающем диалоговом окне "Настройка списка адресов" (рис.1.27) выбрать нужные поля из предлагаемых и добавить недостающие (рис. 1.27, 1.28 ).

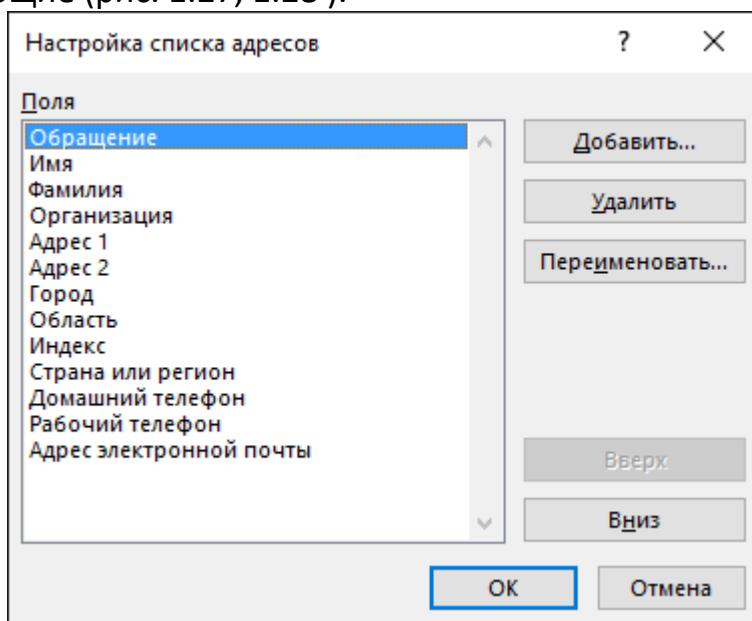


Рис. 1.27.

1. Microsoft Word 2016

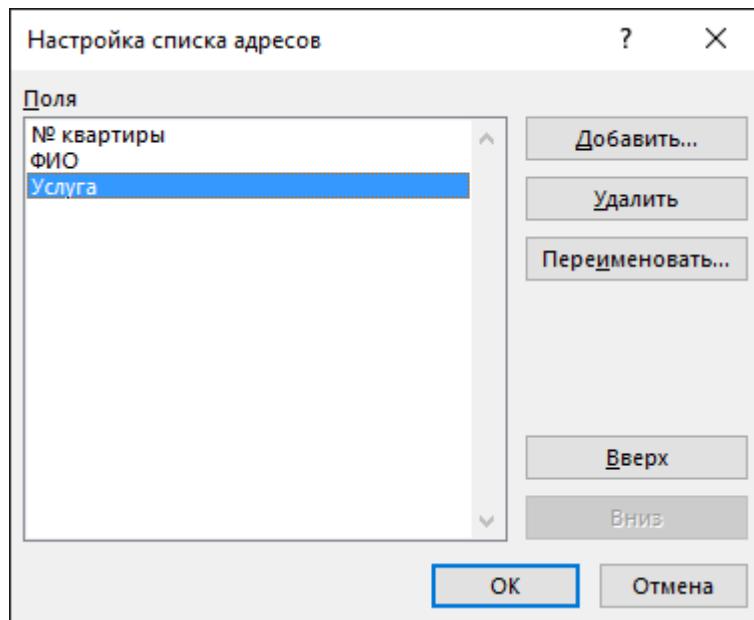


Рис.1.28.

е) в окнах всплывающей панели «Новый список адресов» (рис. 1.29) заполнить поля источника данных;

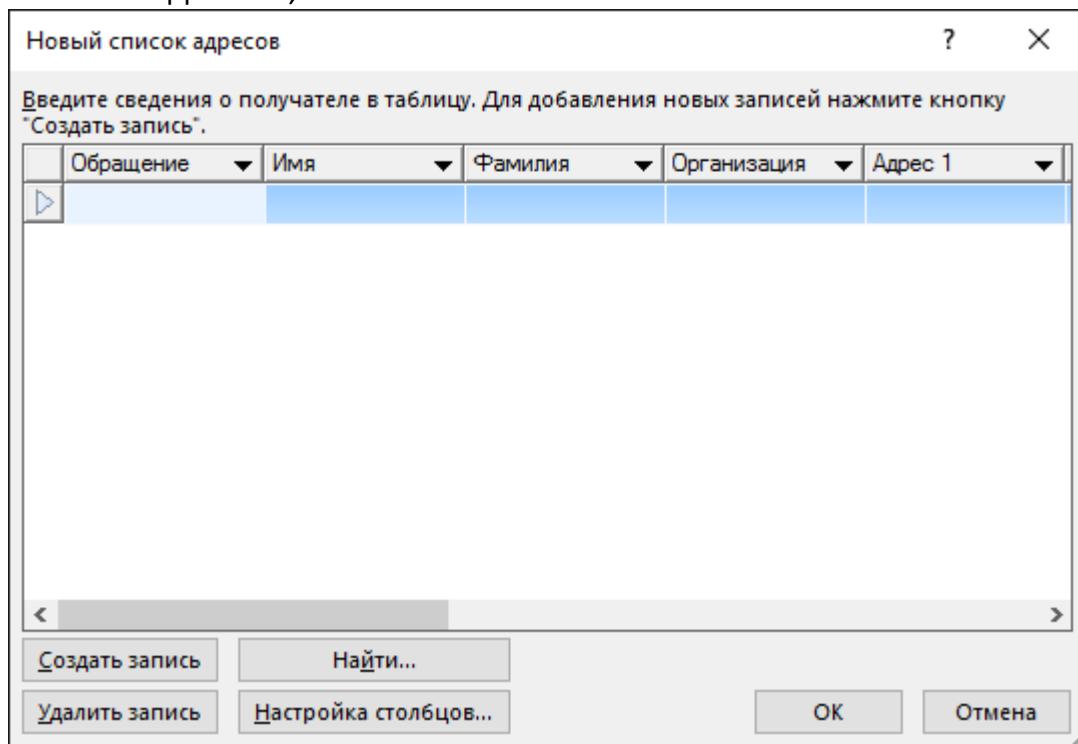


Рис. 1.26.

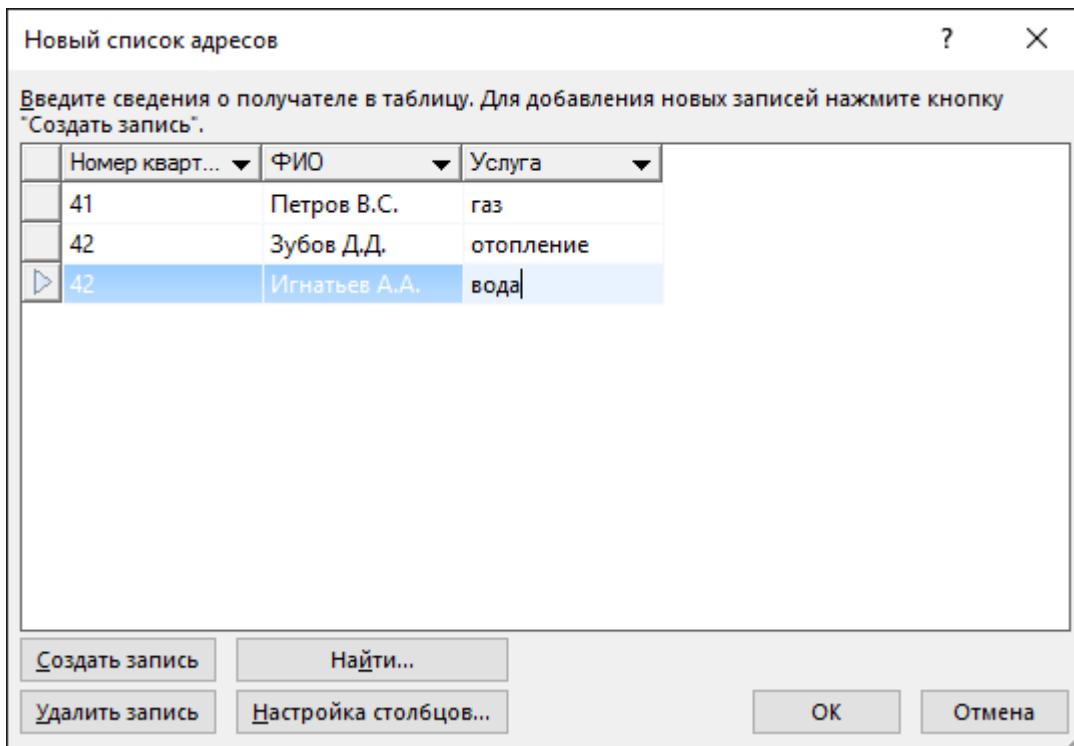


Рис.1.29.

f) сохранить созданную таблицу в файле на диске.

#### ***Создание файла основного документа.***

- a) [Создать сообщение];
- b) создать текст документа, пропуская позиции, которые должны заполняться из файла-источника данных.

#### ***Ввод полей слияния в основной документ***

- a) устанавливать курсор последовательно в позиции слияния - #Рассылки# - !Составление документа и вставка полей! – [Вставить поле слияния];
- b) выбрать из выпадающего списка название поля слияния;
- c) после завершения вставки названий полей слияния [Выделить поля слияния]

В результате в документе появятся затененные названия полей слияния, взятые в кавычки (см. пример ниже).

Ответственному квартиросъемщику  
Квартиры № «Номер\_квартиры»  
Уважаемый товарищ «ФИО» !

Напоминаем Вам, что срок уплаты за «Услуга» истек. Просим срочно произвести оплату.

Председатель ЖСК 1014

Козлов А.М.

### ***Слияние документов***

#Рассылки# - !Просмотр результатов! – [Просмотреть результаты]. В результате поля слияния заполняются данными из файла-источника данных.

Можно, используя элементы прокрутки в группе !Просмотр результатов!, выбрать данные из любой строки файла-источника данных

### ***Пример.***

Ответственному квартиросъемщику  
квартиры № 12

Уважаемый товарищ Петров В.В.!

Напоминаем Вам, что срок уплаты за газ истек. Просим срочно произвести оплату.

Председатель ЖСК 1014

Козлов А.М.

### ***Завершение слияния***

- a) #Рассылки# - !Завершение! – [Найти и объединить];
- b) во всплывающем диалоговом окне «Составные печатные документы» (рис.1.30) указать состав объединенного документа;
- c) сохранить объединенный документ в файле: [Печать документов...]

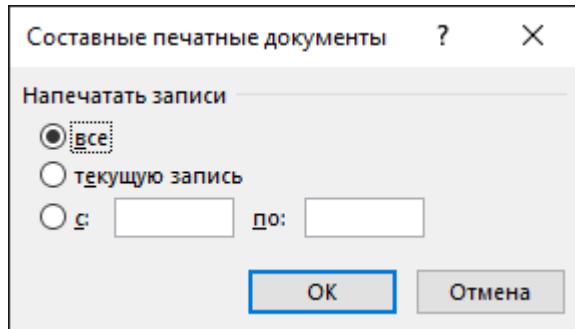


Рис. 1.30.

## ***1.6. Создание гиперссылок***

Для создания гиперссылки:

- выделить текст или рисунок, который должен представлять гиперссылку;
- #Вставка# - !Ссылки! – [Гиперссылка].

Можно также выбрать «Гиперссылка» в контекстном меню выделенного текста или рисунка.

Можно установить связь с уже созданным файлом, местом в документе, веб-страницей, электронной почтой. Для этого в диалоговом окне «Вставка гиперссылки» нужно выбрать соответствующий пункт.

После этого механизм гиперссылок будет работать в редакторе Word.

## ***1.7. Сохранение документа на магнитном диске***

Для записи подготовленного Вами документа на магнитный диск #Файл# - [Сохранить как].

В результате на экране появится диалоговая панель, в которой Вы можете указать имя файла, выбрать папку, в которой будет сохранен файл.

Возможно сохранение файла с защитой от несанкционированного доступа или от записи:

#Файл# - [Сохранить как] - [Сервис] – [Общие параметры] - {Пароль защиты}<sup>8</sup> - [Сохранить].

---

<sup>8</sup> Здесь и далее фигурными скобками указывается текстовый ввод с клавиатуры

### ***1.8. Печать документа***

Для вывода на печать нужно:

#Файл# - [Печать].

До вывода на печать полезно посмотреть расположение текста и рисунков на странице.

### ***1.9. Выход из редактора WORD***

Для выхода из редактора нужно закрыть редактируемый файл:

#Файл# - [Закрыть].

**Приложение. Клавиши быстрого доступа**

Табл. 1.3.

Клавиши	Действие
Ctrl+N	Открыть новое окно ввода
Ctrl+O	Открыть документ
Ctrl+S	Сохранить файл
Ctrl+C	Копировать в буфер хранения
Ctrl+X	Переместить в буфер хранения (Вырезать)
Ctrl+V	Вставить из буфера хранения
Ctrl+Shift+C	Копировать формат
Ctrl+B	Полужирный шрифт
Ctrl+I	Курсив
Ctrl+U	Подчеркивание
Ctrl+=	Подстрочный индекс
Ctrl+Shift++	Надстрочный индекс
Ctrl+Shift+F	Выбор шрифта
Ctrl+Shift+P	Размер шрифта
Ctrl+>	Увеличить размер шрифта
Ctrl+<	Уменьшить размер шрифта
Ctrl+L	Выравнивание по левому краю
Ctrl+E	Выравнивание по центру
Ctrl+J	Выравнивание по ширине
Ctrl+R	Выравнивание по правому краю
Ctrl+*	Отображать скрытые символы форматирования
Ctrl+Z	Отменить действие
Ctrl+Y	Повторить действие
Ctrl+K	Создать гиперссылку
Alt+Ctrl+F	Вставить сноску в конце страницы
Alt+Ctrl+D`	Вставить концевую сноску
Alt+Shift+X	Определение элемента предметного указателя
F7	Проверка орфографии или грамматики
Shift+F7	Вывод синонимов
Ctrl+Shift+E	Исправления документа

## 1.10. Упражнения

### Подготовка, редактирование и печать простых документов

#### 1. Форматирование текстового документа.

Загрузите неформатированный документ из файла article.docx и выполните его форматирование в соответствии со следующими требованиями:

- текст статьи должен быть набран с размером кегля 10, междустрочным интервалом – одинарным;
- вид шрифта – Times New Roman
- размеры полей: верхнего – 2,2 см., нижнего – 1,5 см., левого – 2,0 см., правого – 1,5 см., поле подшивки – 0 см;
- вся статья должна быть сформатирована по ширине, т.е. под полное заполнение строк, переносы слов не допускаются;
- фамилии авторов должны быть выполнены курсивом и выровнены вправо;
- заголовок статьи должен быть выполнен заглавными буквами, полужирным шрифтом, размер шрифта – 12;
- аннотация статьи должна быть выровнена вправо, длина строк – 2/3 от полной.

#### 2. Создание и редактирование рисунков в векторной графике Word.

Создайте документ с рисунком по образцу рис. 1.31, объедините фрагменты рисунка в один графический объект и вставьте рисунок в созданный по п. 1 документ (статью).

#### 3. Вставка формул.

Создайте средствами Microsoft Equation приведенные ниже формулы и вставьте их в созданный по п. 1 документ (статью).

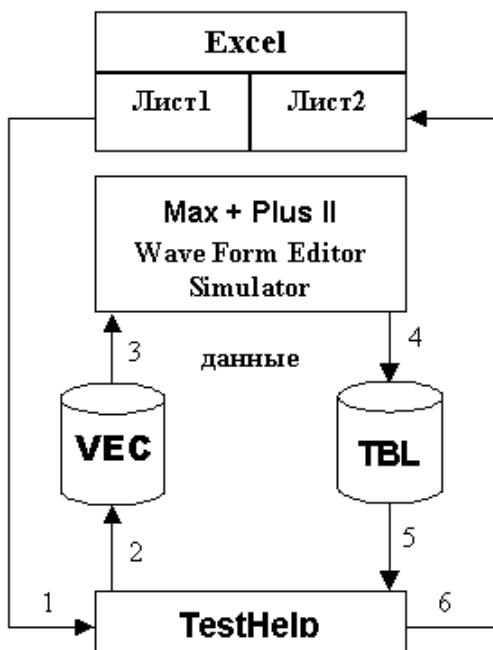


Рис. 1.31. Структура системы программного моделирования процесса обработки

$$S_1 = \sum_{j=1}^m \sum_{i=1}^n Y_{ij};$$

$$S_2 = \sum_{j=1}^m \sum_{i=1}^n Y_{ij} x_{2j};$$

$$S_3 = \sum_{j=1}^m \sum_{i=1}^n Y_{ij} x_{2j}^*;$$

$$S_4 = \sum_{j=1}^m \sum_{i=1}^n Y_{ij} x_{1i} x_{2j};$$

$$S_5 = \sum_{j=1}^m \sum_{i=1}^n Y_{ij} x_{1i};$$

$$S_6 = \sum_{j=1}^m \sum_{i=1}^n Y_{ij} x_{1i}^*.$$

#### 4. Сохранение документа.

Сохраните документ на магнитном диске с защитой от несанкционированного доступа.

#### 2. Создание и форматирование таблиц в документах.

Создайте таблицу по следующему образцу и произведите ее автоформатирование.

АКВАФОР В300

Табл. 1.4.

Ноpp	Показатели воды, мг/дм <sup>3</sup>	Норматив ПДК, мг/дм <sup>3</sup>	Показатели в конце ресурса		Степень очистки, С1/С2
			До очистки С1, мг/дм <sup>3</sup>	После очистки С2, мг/дм <sup>3</sup>	
1	Нитраты	45,0	9,14	9,0	1,1
2	РН	6,0-9,0	7,7	7,8	0,98
3	Аммоний	2,0	0,39	0,35	1,11
4	Нитриты	3,3	0,027	0,025	1,0
5	Хлорофом	0,06	0,06	0,019	3,15

#### 5. Использование стандартных и создание новых шаблонов документов.

- а) создайте текстовый документ на основе одного из стандартных шаблонов резюме, предоставляемых Word;
- б) создайте собственный шаблон документа по следующему образцу :

Приложение №9  
к государственному контракту № \_\_\_\_\_  
от " " г.

**ОТДЕЛЬНАЯ СМЕТА РАСХОДОВ  
ГОЛОВНОГО ИСПОЛНИТЕЛЯ**  
на выполнение работы по проекту № \_\_\_\_\_

---

---

---

(наименование проекта)

№№	Код	Предметные статьи расходов	Сумма, тыс. руб.
1	110100	Оплата труда	
2	110200	Начисления на оплату труда	

затем создайте документ на основе этого шаблона.

**Подготовка и редактирование сложных документов**

Откройте текстовый файл example.docx., содержащий многостраничный неформатированный текст руководства пользователя Word 2016. В результате обработки должен быть получен оригинал-макет для издания руководства, близкий по оформлению к разделу "Microsoft Word 2016" в данном учебном пособии. Для этого произведите операции оформления титульного листа, колонтитулов, вставки и нумерации рисунков, и таблиц, обрамление и заполнение абзацев, оформления заголовков, автоформирование оглавления и предметного указателя, нумерацию и создание списка рисунков и таблиц. Выполните редактирование структуры документа, изменив порядок следования разделов и уровней заголовков в режиме «Структура».

## **Подготовка прикладных документов**

### ***Подготовка финансовых документов.***

1. Создайте по приведенному образцу документ типа "ТОВАРНЫЙ СЧЕТ", используя его бланк в файле plat\_doc.docxx. В бланке заполните вычисляемые поля в столбце "Сумма".
2. Создайте документ типа "ПЛАТЕЖНОЕ ПОРУЧЕНИЕ", используя его бланк в файле plat\_doc.docx. В бланке создайте поля со списком в клетках строки "Получатель" и "Банк получателя".

### ***Создание документов путем слияния.***

Создайте по приведенному ниже образцу путем слияния серию документов типа извещения квартиросъемщиков об уплате за услуги (горячую воду, газ, лифт и т.д.)

*Ответственному квартиросъемщику квартиры № «Номер\_квартиры». Уважаемый товарищ «Обращение»!*

*Напоминаем Вам, что срок уплаты за «Услуга» истек. Просим срочно произвести оплату.*

*Председатель ЖСК 1234*

*Козлов К.К.*

## **Подготовка рекламных листков**

Создайте рекламный листок по образцу, рис. 1.32 (с одноколоночным заголовком, двухколоночным расположением текста), на основе исходного текстового файла adc.docx и кадров изображений из графических файлов adc402.bmp и VI\_panel.bmp.

# ADC-402

Двухканальный цифровой запоминающий осциллограф



## Функциональное назначение

Модуль ADC-402 представляет собой функционально законченный двухканальный цифровой запоминающий осциллограф, выполненный в виде платы, устанавливаемой в IBM PC/AT совместимый компьютер. Использование программного обеспечения модуля ADC-402 позволяет использовать компьютер как двухканальный запоминающий осциллограф с функциями фильтрации и накопления сигнала, спектральный анализатор, коррелятор. Интерфейс программ, выполненный в виде традиционного осциллографа, позволяет легко пользоваться модулем ADC-402 потребителю, знакомому с работой стандартного осциллографа.

Дополнительные возможности модуля ADC-402 расширены цифровым портом ввода / вывода для управления управление аппаратурой автоматизации эксперимента. Для этих целей в составе модуля ADC-402 предусмотрены 24 цифровых линии ввода / вывода, программируемые в стандарте микросхемы ППИ 18255 фирмы Intel.

Гибкая система синхронизации, задаваемая программно, позволяет реализовать различные режимы работы и исследовать сигналы, как синхронные с моментом начала преобразования, так и случайные. Система автокалибровки позволяет проводить периодическую цифровую коррекцию передаточной характеристики ADC-402.

Возможность сохранять "осциллограммы" сигналов позволяет решать задачи идентификации.

## Характеристики

- 2 синхронных канала ввода аналоговых сигналов
- максимальная частота дискретизации - 40 МГц
- биполярный и однополярный диапазон входных сигналов
- 8-ми разрядное (ADC-402) или 10-ти разрядное (ADC-40210) разрешение
- расширенная система синхронизации от аналоговых и цифровых сигналов
- 64 Кб буферное ОЗУ в каждом канале
- шина ISA
- система обмена данными через порты ввода /вывода IBM PC/AT
- поддержка работы с прерываниями

- 24 цифровых линии ввода /вывода

## Применение

- высокоскоростное аналого - цифровое преобразование
- анализ электронных схем и компонентов
- стандартные измерения
- тестирование продукции
- медицинские исследования
- спектральный анализ
- оборудование измерения вибраций

## Программирование

Аппаратная часть модуля ADC-402 построена таким образом, что его можно программировать для работы в системах WINDOWS/2000,XP. Программирование модуля ADC-402 возможно в средах высокого уровня C/C++, VISUAL C++, VISUAL BASIC и специализированных программных оболочках, например, Lab WINDOWS/CVI фирмы National Instruments.

В комплект поставки программного обеспечения модуля ADC-402 входят:

- \* исходные тексты библиотеки функций управления модулем ADC-402 на языке C++
- \* исходные тексты программ виртуальных приборов - накопителя, коррелятора и запоминающего осциллографа
- \* программа тестирования работоспособности модуля ADC-402

Вид одного из рабочих экранов программы «ДВУХКАНАЛЬНЫЙ ЗАПОМИНАЮЩИЙ ОСЦИЛЛОГРАФ»

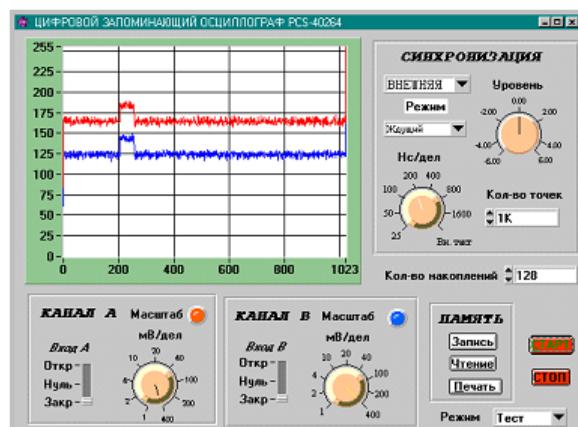


Рис. 1.32.

## 2.Excel 2016

Электронная таблица Microsoft Excel 2016 позволяет автоматизировать процесс создания и редактирования табличных (например, бухгалтерских) документов, предоставляя функции Автозаполнения, Автосуммирования, Автоформирования графиков и диаграмм, Автовычисления по формулам, предоставляет возможность работать с таблицей как с базой данных, решать задачи анализа данных, подбора параметров и оптимизации.

### 2.1. Запуск Excel 2016

Запуск программы Excel 2016 в WINDOWS 10/8/7 можно произвести из главного меню:

[Пуск] - [Программы] - [Microsoft Office] – [Microsoft Office Excel 2016 ]  
В результате откроется окно Excel 2016 и будет загружена новая рабочая книга Книга1. Элементами окна Excel 2016 (Рис. 2.1) являются :

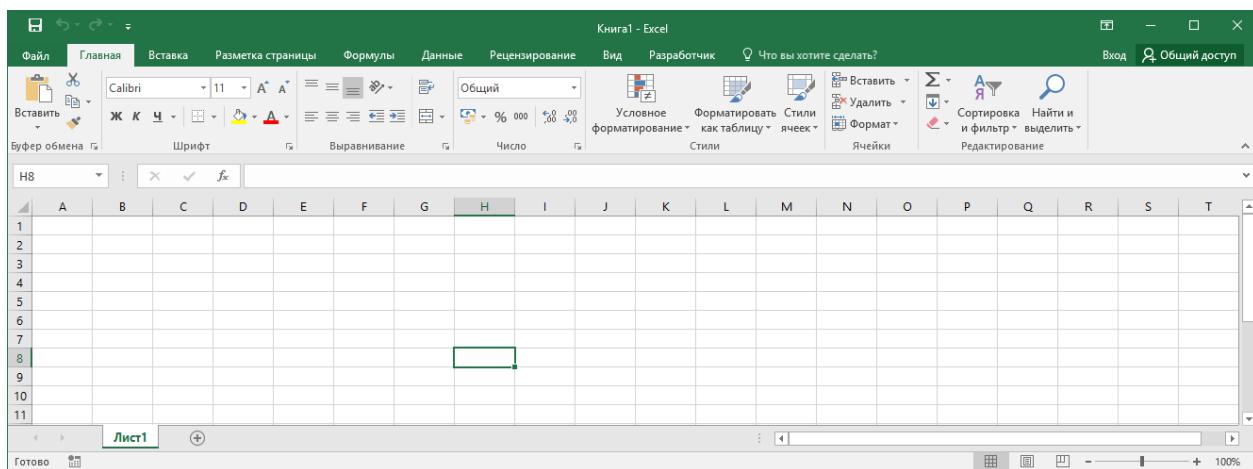


Рис. 2.1.

- строка заголовка;
- размерные кнопки - «Развернуть» и «Свернуть»;
- вкладки с группами пиктограмм (главное меню);
- панель быстрого доступа;
- строка формул;
- рабочая область таблицы с заголовками столбцов и строк.
- линейки прокрутки;
- ярлычки листов и кнопки прокрутки ярлычков;
- статусная строка;

- область задач.

Для удобства работы с редактором можно раскрыть окно на весь экран, сделав щелчок мышью на размерной кнопке «Развернуть» в строке заголовка.

После этого можно формировать табличный документ с клавиатуры или загрузить в окно текстовый файл с магнитного диска. Выбор действий в Excel 2016 производится с помощью главного меню, пиктограмм на панели быстрого доступа, контекстного меню и меню смарт-тегов<sup>9</sup>. Основным инструментом пользователя при работе в среде Excel 2016 является графический манипулятор мыши. Выполняемая функция определяется формой, которую приобретет курсор мыши, и тем, какую клавишу мыши Вы нажимаете и каким образом.

## Формы курсора мыши

- курсор ожидания выполнения действия (песочные часы). Если Вы видите такой курсор - надо подождать завершения действия;
- курсор выделения ячейки или группы ячеек ("санитарный крест");
- курсор автозаполнения (крест);
- курсор перемещения (стрелка, наклоненная влево);
- курсор редактирования (I);
- курсор изменения размеров окна ( $\leftrightarrow$ );
- курсор изменения границ ( $\leftrightarrow$ );
- курсор разделения на подокна;
- курсор Мастера диаграмм.

---

<sup>9</sup> Смарт-тег – это пиктограмма, появляющаяся после в процессе выполнения некоторых действий. Активизация смарт-тега с помощью мыши приводит к появлению дополнительного меню режима.

## **2.2.Создание таблицы**

Лист рабочей книги, часть которого видна на экране, разделен на клетки (ячейки).

Полное количество столбцов - 16384, строк - 1048576. В любую клетку Вы можете вводить текст, числовые данные или формулу произвольной длины.

Рабочая книга содержит 3 листа. Количество листов в рабочей книге может быть увеличено пользователем и ограничено только ресурсами памяти компьютера.

### **Ввод текста, цифровых данных и формул в клетку (ячейку) таблицы**

Для ввода цифровых данных и формул в клетку (ячейку) таблицы нужно:

а) активизировать ячейку, установив на нее курсор выделения ячейки и сделав щелчок левой кнопкой мыши;

б) ввести текст, цифровые данные или формулу. Формула должна начинаться со знака "=". В формуле могут использоваться числовые величины, адреса ячеек таблицы<sup>10</sup>, знаки арифметических операций, стандартные функции Excel 2016. После окончания ввода нужно нажать клавишу <Enter><sup>11</sup>. Можно вместо <Enter> нажать “стрелку вниз, вправо, влево или вверх” При этом будет не только закончен процесс ввода, но и активизирована следующая клетка.

Процесс формирования таблицы может быть ускорен благодаря наличию функций Автозаполнения и Автосуммирования.

При вводе в формулу адреса ячейки, находящейся на другом листе таблицы, достаточно сделать щелчок левой кнопкой мыши на ярлычке этого листа и затем - на адресуемой ячейке.

В ячейку с помощью Мастера функций может быть внесена стандартная формула Excel 2016 из категорий: финансовые, математические, статистические и др. Кнопка вызова Мастера функций находится в строке формул.

Создание формул и работу с ними можно выполнить, используя пиктограммы, представленные на вкладке «Формулы».

---

<sup>10</sup> Адрес ячейки может быть относительным (A1, B5 и.т.д.), абсолютным (\$A\$1, \$B\$6). Можно адресовать ячейку на другом листе рабочей книги (<имя листа>!<адрес ячейки>, например, Лист1!A1, БД!F4)

<sup>11</sup> Здесь и далее обозначение с названием клавиши, взятым в треугольные скобки обозначает действие нажатия соответствующей клавиши.

## Автозаполнение

*Если клетки одной строки содержат одинаковые данные или содержат элементы списка, например, названия месяцев или дней недели, нужно:*

- заполнить первую клетку;
- активизировать ее;
- произвести автозаполнение строки.

Для этого найти в правом нижнем углу клетки курсор автозаполнения (см. рис. 2.2), затем, нажав и удерживая нажатой левую кнопку мыши, провести курсор по строке до последней заполняемой клетки. Если провести курсор при нажатой клавише Ctrl, то в строку (столбец) будет введен возрастающий числовой ряд.

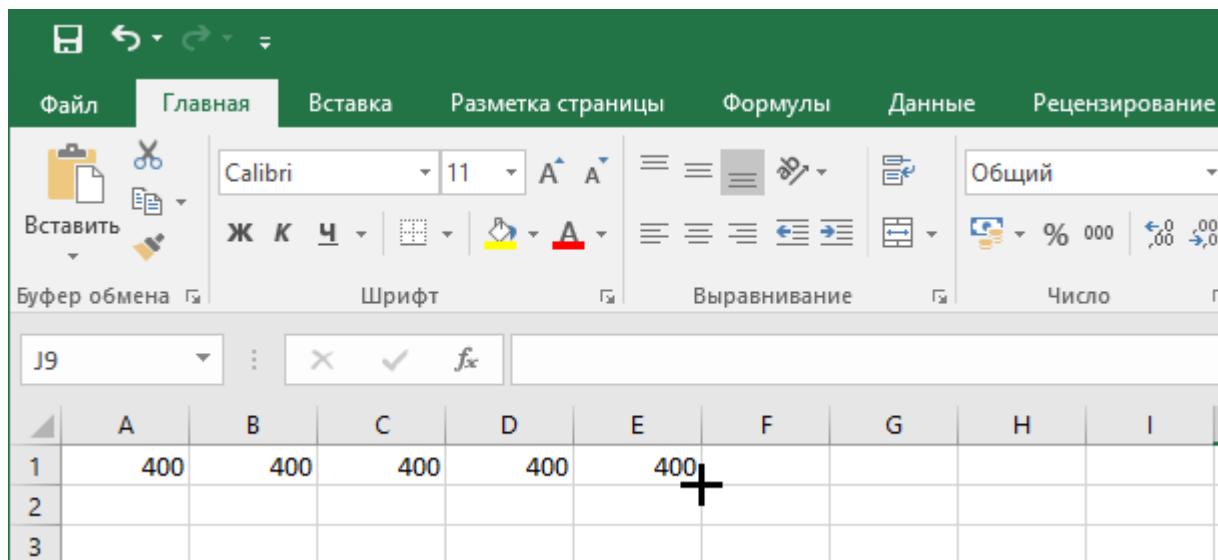


Рис. 2.2.

Если требуемого списка нет, можно его создать. для этого нужно последовательно выполнить следующие действия:

- #Файл# – [Параметры] – [Основные] - [Изменить списки] – [Добавить];
- ввести элементы списка, нажимая клавишу <Enter> после ввода каждого элемента.

*Если клетки одной строки представляют регулярный числовой ряд:*

**1) арифметическую прогрессию:**

- заполнить первые две клетки ;
- произвести пометку этих двух клеток.

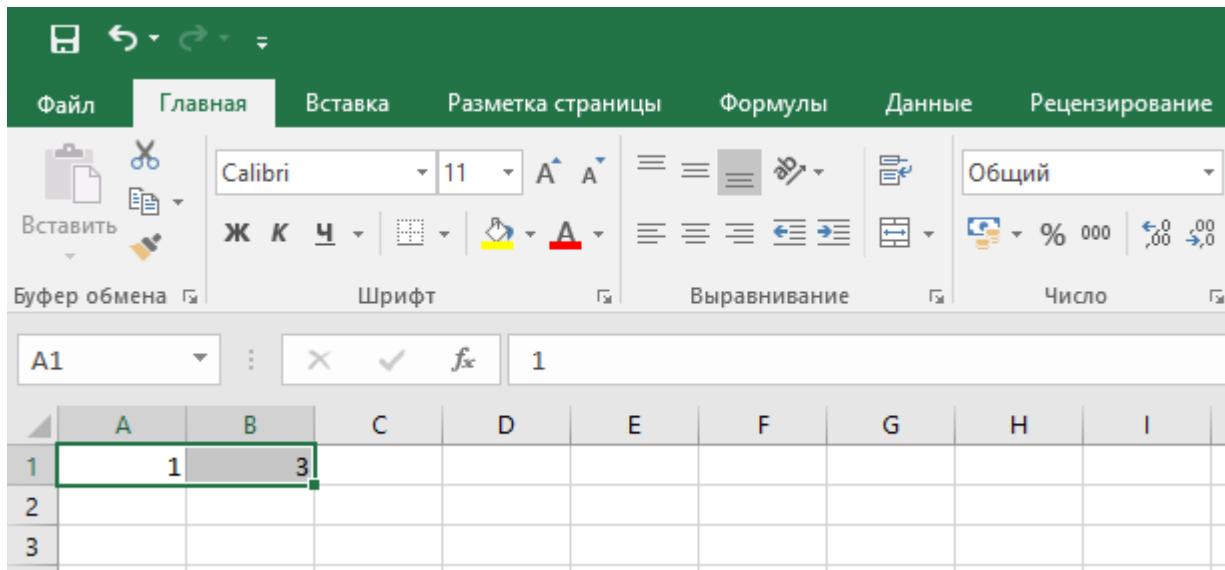


Рис. 2.3.

Для пометки двух клеток нужно активизировать первую клетку, установить курсор выделения ячейки на эту клетку, нажать левую кнопку мыши и, удерживая ее нажатой, провести курсор до второй клетки (см. рис. 2.3).

с) произвести автозаполнение клеток строки, в которых должна находиться прогрессия. Для этого нужно установить курсор мыши в правый нижний угол выделенной области (в результате курсор примет вид крестика), нажать левую кнопку мыши и, удерживая ее нажатой, провести курсор до конечной клетки (см. рис. 2.4). Используя дополнительное всплывающее меню (см. рис. 2.4) можно не только заполнить ячейки, но также копировать ячейки, заполнить только форматы, заполнить только значения.

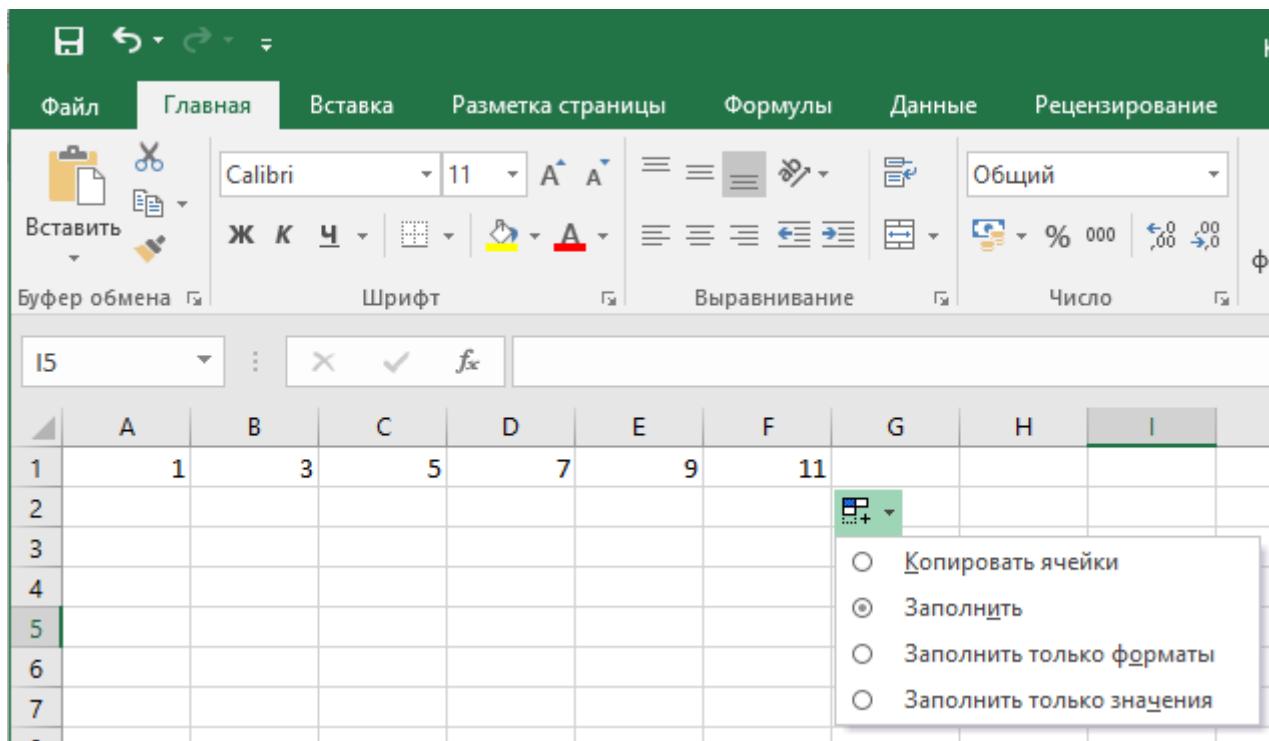


Рис. 2.4.

## 2) геометрическую прогрессию:

- заполнить первую клетку;
- пометить ряд клеток в строке или столбце, в которых должна быть прогрессия;
- #Главная# - !Редактирование! - [Заполнить] - [Прогрессия...];
- выбрать тип прогрессии «геометрическая», задать шаг, конечное значение во всплывающем диалоговом окне «Прогрессия» (рис. 2.5).

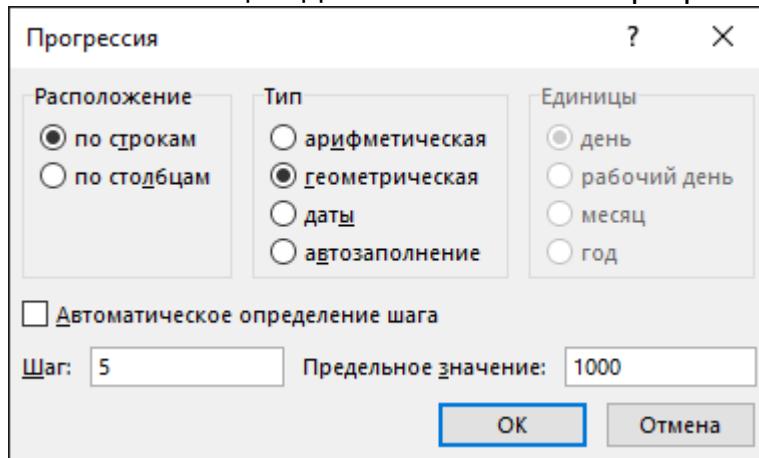


Рис. 2.5.

## Автосуммирование

Позволяет автоматически вычислять строку (столбец) с итоговыми суммами.

Для выполнения автосуммирования нужно:

- пометить пустую строку (столбец), следующую снизу (справа) за полем с цифровыми данными;
- #Формулы# - !Библиотека функций! – [Автосумма] – [Сумма]

Используя дополнительное меню «Автосумма», можно использовать и другие функции (см. рис. 2.6).

			D	E	F	G
1			125	625		
2	4	4	4	4	4	
3	5	9	29	129	629	
4						

Рис. 2.6.

## 2.3. Редактирование таблицы

При редактировании можно:

- дополнить таблицу;
- прозвести необходимые изменения текста, формул или цифровых данных в любой клетке;
- изменить шрифт, стиль написания и размер символов в любом фрагменте таблицы;
- произвести изменение размеров строк и столбцов или автоподстройку с учетом содержащегося в них текста;
- произвести автоформатирование таблицы в целом;
- произвести автоматическое задание процентного или денежного стиля представления данных в требуемой области таблицы, увеличить или уменьшить разрядность данных.

Можно также отменять предыдущую операцию редактирования.

Операции редактирования (удаление, копирование, перемещение фрагментов текста (слова, строки, строчного или прямоугольного блока текста)) производятся после того как данный фрагмент текста помечен.

## **Пометка элементов таблицы**

Можно пометить:

- отдельную клетку, если установить на нее курсор выделения ячейки и сделать щелчок левой кнопкой мыши;
- столбец (строку), если сделать щелчок левой кнопки мыши на заголовке столбца (строки);
- прямоугольную область ячеек, если установить курсор выделения ячейки на ячейку в верхнем левом углу области, затем нажать левую кнопку мыши и, удерживая ее нажатой, провести курсор мыши до правой нижней клетки области;
- несмежные области, если в процессе пометки удерживать <Ctrl>.
- большую прямоугольную область ячеек, если:
  - а) выделить ячейку в левом верхнем углу области;
  - б) #Главная# - !Редактирование! – [Найти и выделить] - [Перейти...];
  - в) ввести в окно «Ссылка» панели «Переход» адрес ячейки в правом нижнем углу области;
  - г) <Shift> + [OK] .

Второй способ – выделить ячейку в левом верхнем углу области (таблицы), затем нажать клавишу <Shift> и сделать щелчок левой кнопкой мыши на ячейке в правом нижнем углу области (таблицы).

## **Редактирование ячейки**

- а) пометить ячейку (клетку). В результате ее содержимое (текст, цифровые данные или формула) отобразится в строке формул;
- б) установить курсор редактирования в строку формул и зафиксировать курсор в этой строке щелчком левой кнопки мыши;
- в) выполнить редактирование в строке формул, после окончания редактирования - <Enter>.

Можно выполнять редактирование непосредственно в ячейке. Для этого нужно активизировать ячейку, установить курсор на активированной ячейке и сделать двойной щелчок левой кнопкой мыши.

### **Вставка, копирование, перемещение фрагмента таблицы**

- a) пометить фрагмент таблицы;
- b) занести помеченный фрагмент в буфер хранения:
- c) #Главная# - !Буфер обмена! - [Копировать] - если нужно копировать или  
#Главная# - !Буфер обмена! - [Вырезать] - если нужно перенести;
- d) переместить курсор выделения ячейки в левый верхний угол места копирования или вставки (в пределах одного листа, на другой лист, в другое окно);
- e) вставить или скопировать фрагмент таблицы:  
#Главная# - !Буфер обмена! - [Вставить].

Действия копирования, вырезания и вставки может быть выполнены с помощью комбинации клавиш клавиатуры <Ctrl+C>, <Ctrl+X>, <Ctrl+V> соответственно.

#### *Изменение шрифта, стиля написания, размера символов.*

Для этого необходимо пометить фрагмент таблицы, затем нужно последовательно выбрать пункты главного меню:

#Главная# - !Шрифт! – [Открыть диалоговое окно]

после чего произвести установки, описанные в п.3.

Некоторые изменения (введение жирного шрифта, курсива, подчеркивания ) можно сделать проще, используя соответствующие пиктограммы #Главная# - !Шрифт!.

### *Вставка, копирование, перемещение фрагмента таблицы методом буксировки*

- а) пометить фрагмент таблицы;
- б) найти курсор перемещения (в любом месте контура помеченной области);

в) для перемещения фрагмента таблицы нужно нажать левую кнопку мыши и, не отпуская ее, произвести буксировку. Для копирования фрагмента таблицы нужно произвести буксировку при нажатой клавише <Ctrl>.

## **Форматирование таблицы**

До начала формирования таблицы можно произвести форматирование символов, т.е. установить необходимый шрифт, вид шрифта (стиль написания) и размер символов, используя элементы управления на вкладке «Шрифт», например: Шрифт: Calibri , Начертание: курсив , Размер: 12.

### **2.4.Форматирование таблицы**

Под форматированием понимается задание вида и размера шрифта, расположения текста или данных в клетках, а именно: отступ слева, справа, а также выравнивание текста в клетках по левому краю, по правому краю, по центру. Для этого:

- а) пометить фрагмент таблицы;
- б) #Главная# - !Шрифт! – [Открыть диалоговое окно];
- с) задать параметры формата на вкладках «Число», «Выравнивание», «Шрифт» и других во всплывающей панели (рис. 23). В частности, можно задать в клетке перенос текста по словам, установив флажок «переносить по словам» на вкладке «Выравнивание».

Специальный вид форматирования ячейки – условное форматирование. С помощью этого приема можно задать особые параметры формата (например, другой цвет) в ячейках, отвечающих какому-либо условию, например, если значения превышают заданный порог. Для введения условного форматирования нужно:

- а) пометить ячейки таблицы;
- б) #Главная# - !Стили! – [Условное форматирование];
- с) задать условия и вид форматирования.

Можно задать примечания к ячейкам. Для этого нужно:

- а) пометить ячейку.
- б) вызвать щелчком правой кнопки мыши контекстное меню.
- с) [Вставить примечание];
- д) ввести текст примечания.

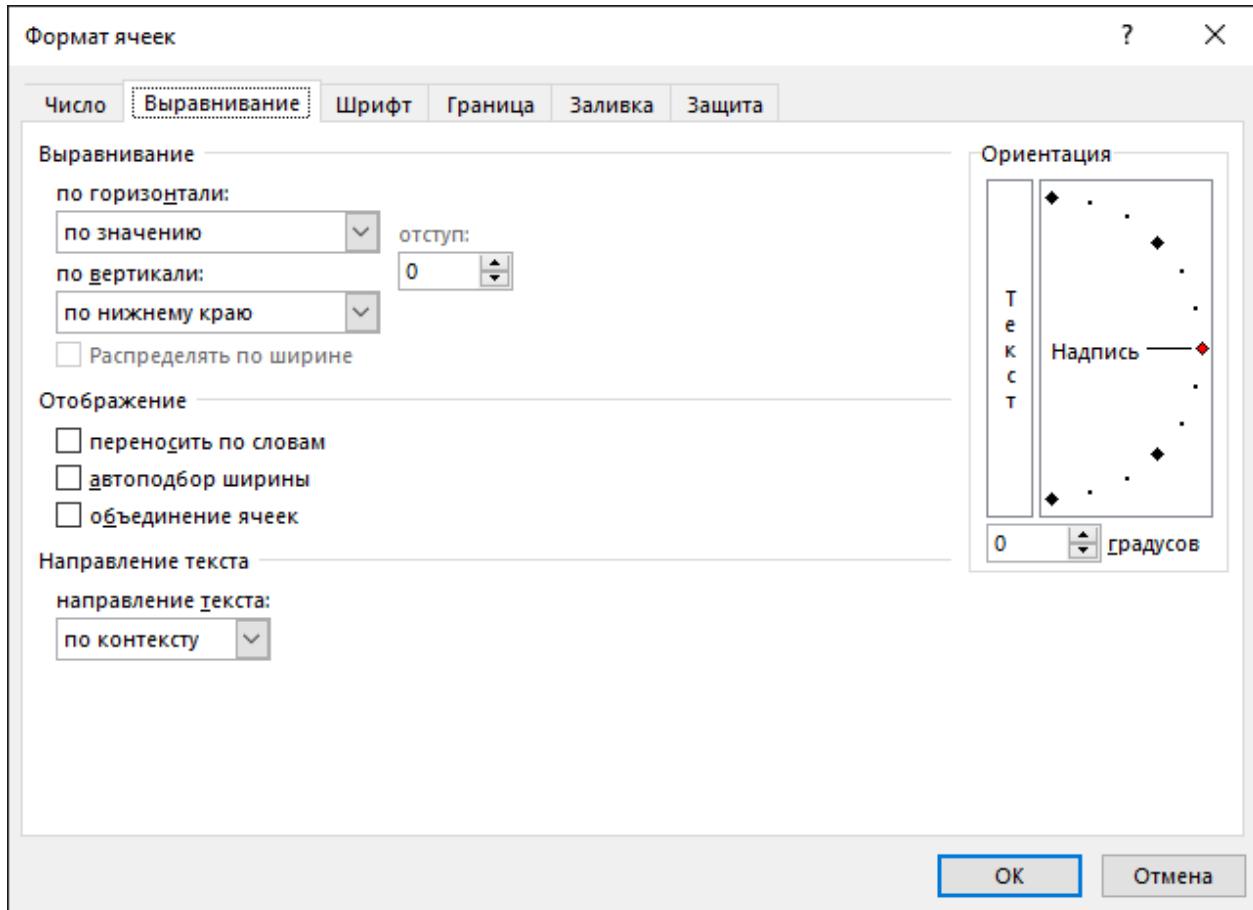


Рис. 2.7.

### ***Форматирование столбцов (строк)***

Форматирование строк(столбцов) включает:

Настройку высоты строк (ширины столбцов):

#Главная# - !Ячейки! – [Формат] – [Высота строк]/[Ширина столбцов]

Автоподбор высоты строк (ширины столбцов):

#Главная# - !Ячейки! – [Формат] – [Автоподбор высоты строк]/[Автоподбор ширины столбцов]

### ***Форматирование таблицы в целом***

При форматировании таблицы в целом выбирается один из стандартных видов внешнего оформления таблицы. Для этого:

- пометить форматируемую область таблицы;

- b) #Главная#- !Стили! - [Форматировать как таблицу];
- c) выбрать из выпадающего меню стиль таблицы.

Можно скрыть сетку в таблице:

- a) #Разметка страницы# - !Параметры листа! - [Сетка];
- b) сбросить флажок «Показать».

### ***Копирование формата***

- a) выделить ячейку или группу ячеек, формат чисел или текста из которых нужно скопировать;
- b) #Главная# - !Буфер обмена! – [Формат по образцу];
- c) выделить ячейку или группу ячеек, в которые нужно скопировать формат. После того как пометка ячеек будет закончена, произойдет копирование формата. Можно произвести копирование формата столбцов/строк таблицы.

### ***Введение форматов чисел***

- a) пометить область таблицы;
- b) #Главная# – !Число! – [Финансовый числовой формат]/[Процентный формат];
- c) выбрать вид денежной единицы, если нужно ввести денежный формат.

В Excel 2016 предусмотрены и другие форматы чисел: дробный, экспоненциальный, краткий формат даты и др. Для задания одного из этих форматов:

- a) пометить область таблицы;
- b) #Главная# – !Число! ;
- c) выбрать вид формата из поля со списком или/и с помощью пиктограмм на вкладке «Число».

### ***Изменение разрядности данных***

- a) пометить строку, столбец или область, в которой нужно изменить разрядность;

- b) #Главная# - !Число! – [Уменьшить разрядность]/[Увеличить разрядность].

### **Отмена последней операции редактирования**

Для отмены последней операции редактирования нужно щелкнуть левой кнопкой мыши на пиктограмме «Отменить ввод», находящейся на панели быстрого доступа, или нажатием клавиш <Ctrl+z>.

*Примечание: повторяя отмену можно отказаться от нескольких (до 100) ранее выполненных операций редактирования.*

### **Загрузка файла в окно редактирования**

Если файл таблицы был создан ранее, то его можно загрузить в окно редактирования и продолжить работу с ним.

Для этого нужно щелкнуть левой кнопкой мыши на кнопке «Файл» и выбрать файл из выпадающего списка «Последние документы» или #Файл# – [Открыть] и во всплывающем диалоговом окне выбрать имя папки и загружаемого Вами файла. Можно загрузить последовательно два или более файлов. Для того, чтобы увидеть их одновременно в окне редактора Excel 2016, нужно:

#Вид# - !Окно! - [Упорядочить все] – [<способ расположения>].

### **2.4. Создание и редактирование диаграмм**

Цифровые данные, представленные в таблице Excel 2016, можно отобразить в виде графиков и диаграмм. По оси X могут быть отложены любые текстовые обозначения (например, названия месяцев из шапки таблицы), а по оси Y – данные, содержащиеся в одной или нескольких строках данных таблицы.

### **Создание диаграмм**

- пометить интервал ячеек (строку) с текстовыми подписями для оси X;
- нажать <Ctrl> и, удерживая ее нажатой, пометить первую и другие строки с данными, включая и клетки с названиями строк (см. пример на рис. 2.8);

- c) #Вставка# – !Диаграммы! ;  
d) выбрать вид диаграммы;

		Наименование изделия (пломбы)				
Затраты в себестоимости		Фликлок	Унисил	НП	Нейлсил	Трос
1	Сырье и материалы	1,55	1,07	1,14	21,27	28,75
2	Топливо	0,34	0,24	0,26	2,35	3,39
3	Энергия	0,89	0,62	0,69	0,16	0,23
4	Расходы на оплату труда	0,05	0,03	0,04	0,35	0,51
5	Начисления на зарплату	0,01	0,01	0,01	0,07	0,1
6	Административные расходы	0,35	0,24	0,27	0,42	0,61
	Выплаты, включаемые в себестоимость					
7		0,18	0,12	0,14	1,25	1,8
8	Сбытовые расходы	0,54	0,37	0,42	3,74	5,4
9	Реклама	0,04	0,03	0,03	0,28	0,41
10	Налоги в себестоимости	1,26	0,87	0,97	6,72	9,71
11	Амортизация оборудования	0,04	0,03	0,03	0,06	0,09
12	<b>Себестоимость</b>	<b>5,25</b>	<b>3,63</b>	<b>4</b>	<b>36,67</b>	<b>51</b>

Рис. 2.8

Пример диаграммы приведен на рис. 2.9.

Примечание: при необходимости отобразить зависимость одной величины от другой нужно пометить две строки (столбца) с данными и использовать вид графика XY.

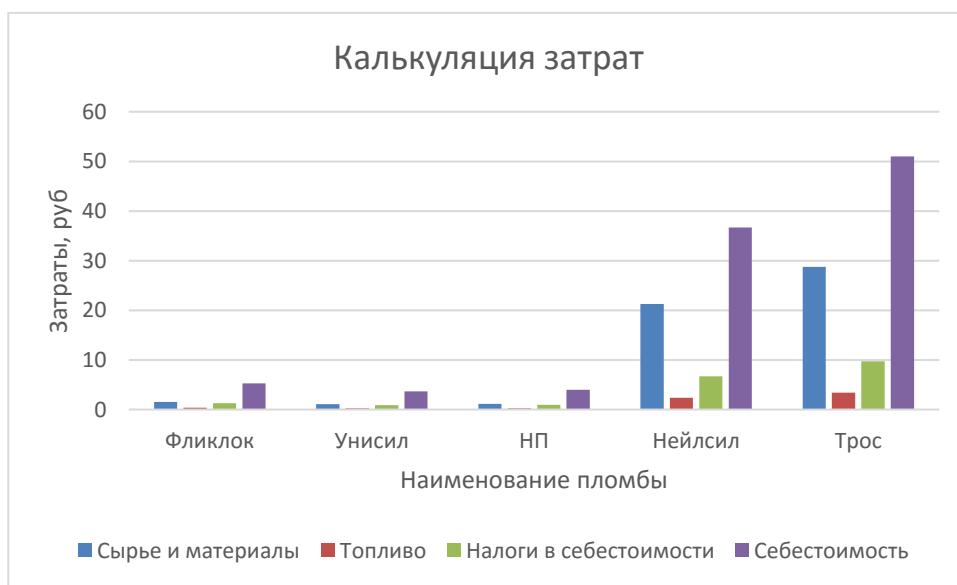
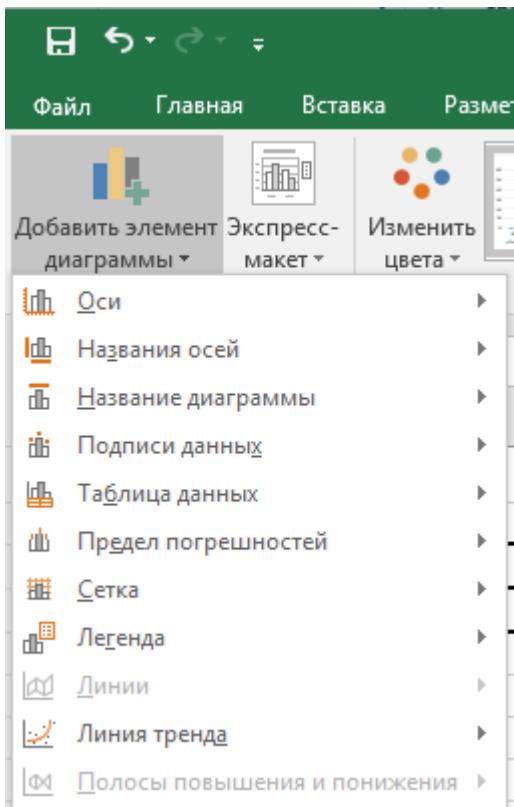


Рис. 2.9.

## Редактирование и анализ диаграмм

Средства редактирования и анализа диаграмм представлены на вкладке «Конструктор».

Можно изменять размеры окна диаграммы, перемещать диаграмму по листу, вводить текстовые данные. Вход в режим редактирования - щелчком мыши на поле диаграммы, выход - щелчком мыши вне поля диаграммы.



Можно добавить элемент созданной диаграммы: заголовок, подписи осей, легенду и др. Для этого:

- #Конструктор# - !Макеты диаграмм! – [Добавить элемент диаграммы];
- выбрать элемент из поля со списком (рис.2.10).

Элементом диаграммы может быть также линия линейного или экспоненциального тренда, прогнозирования и фильтрации.

Рис. 2.10.

## 2.5. Сохранение документа на магнитном диске

Для записи созданного документа на магнитный диск:

#Файл# - [Сохранить как]

В результате на экране появится диалоговая панель, в которой нужно выбрать имя папки и задать имя файла.

Окончание установок - щелчок мышью на кнопке «Сохранить» или нажатие клавиши <Enter>.

Возможно сохранение файла с защитой от несанкционированного

доступа или от записи. Для этого нужно последовательно выбрать пункты текстового меню

#Файл# - [Сохранить как] - [Сервис] – [Общие параметры] - {Пароль защиты}.

## **2.6.Подготовка к печати**

До печати многостраничного документа необходимо проверить правильность задания формата страницы: отступы слева, справа, сверху, снизу, - и откорректировать их при необходимости. Кроме того, полезно просмотреть результат автоматической разбивки на страницы. Если Вас не устраивает предлагаемая разбивка (например, если название какого-нибудь раздела документа оказалось в конце страницы, а нужно, чтобы оно было в начале) - ее можно изменить.

До вывода на печать полезно посмотреть и при необходимости отрегулировать расположение таблиц на страницах и размеры полей. Для этого нужно последовательно выбрать пункты:

#Файл# – [Печать]

В результате на экране будет представлено окно предварительного просмотра, в котором будет показано размещение таблиц на страницах при печати. Для редактирования таблиц нужно выйти из режима предварительного просмотра. На рабочем листе после этого будут показаны не только сами таблицы, но и границы страниц (пунктирными линиями).

## **2.7.Выход на печать**

Для вывода на печать:

#Файл# - [Печать]- [Печать].

## **2.8.Базы данных**

Таблица с заголовками столбцов, размещеными в одной строке, может использоваться в качестве базы данных. Для работы с таблицами, имеющими такую структуру, предусмотрены удобные средства просмотра, добавления и обновления записей, средства сортировки записей и средства поиска и отбора записей по задаваемому пользователем критерию.

### **Сортировка**

- a) пометить таблицу вместе с заголовками столбцов;
- b) #Данные# - [Сортировка и фильтр] - [Сортировка];
- c) во всплывающем диалоговом окне «Сортировка» указать столбец, в котором должна производиться сортировка, и задать ключ сортировки.

### **Поиск в базе данных по нескольким критериям**

- a) пометить таблицу включая заголовки столбцов;
- b) #Данные# - [Сортировка и фильтр] - [Фильтр];
- c) задать условия поиска в нужных столбцах (рис. 26).

Рис. 2.11.

При выборе «Настраиваемый фильтр...» можно задавать условия с ограничениями (рис. 2.12):

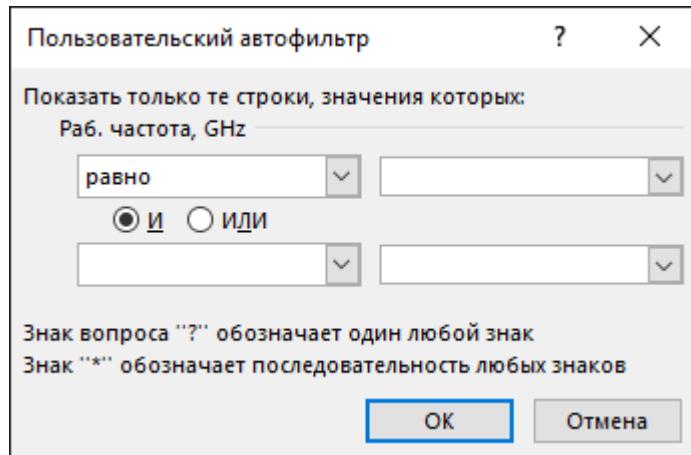


Рис. 2.12.

Результаты поиска можно скопировать на свободную область листа, на другой лист или в другую книгу.

Для того, чтобы вернуть исходный вид таблицы:  
#Данные# - !Сортировка и фильтр! - [Фильтр].

## 2.9.Анализ данных

Анализ данных рассмотрим на примере базы данных (табл. 2.1) о сбыте продукции в течении месяца фирмой-производителем нескольким фирмам-потребителям.

Табл. 2.1.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Дата	Наим.орг.	Кол1	Цена	Сумма	Кол2	Цена	Сумма	Всего	Дата оплаты	Оплата	Остаток
2	01.янв	ВИЛАРД	50	2,45	122,5			0	122,5	15.янв	100	22,5
3	01.янв	КАРАТ	80	2,4	192			0	192			192
4	01.янв	КОНДАС	50	2,45	122,5	20	4,4	88	210,5	05.янв	25	185,5
5	01.янв	ЛИГА	20	2,5	50			0	50			50
6	02.янв	ЛОРА	100	2,6	260	10	4,3	43	303	07.янв	200	103
7	02.янв	СОЛО	50	2,3	115			0	115			115
8	02.янв	ТИГ	50	2,5	125			0	125			125
9	03.янв	АЛЬЯНС	50	2,5	125			0	125			125
10	03.янв	АННА-МАРИЯ	200	2,4	480			0	480			480
11	03.янв	АЯКС	50	2,3	115			0	115			115
12	03.янв	ВИЛАРД	100	2,3	230			0	230			230
13	03.янв	КОНДАС	100	2,5	250			0	250			250
14	03.янв	СТО	50	2,45	122,5			0	122,5			122,5
15	03.янв	ТИГ	100	2,3	230			0	230	10.янв	120	110
16	04.янв	ОЛФОЮ	50	2,45	122,5			0	122,5			122,5

Анализ данных в Excel 2016 позволяет автоматизировать процесс подведения итогов по категориям, например, по фирмам-потребителям:

- количество товара, отгруженного каждой фирме за месяц;
- стоимость отгруженного товара за месяц;
- полученные за поставленный товар денежные суммы за месяц;
- остаток невыплаченных средств за месяц

и по датам. В этом случае требуется отобразить количество денежных средств, полученных фирмой-производителем по датам.

В исходной таблице (базе данных) должны быть представлены необходимые для этого анализа данные:

- дата;
- наименование фирмы - потребителя;
- количество отгруженного товара, может быть, по категориям;
- стоимость отгруженного товара;
- количество денежных средств, выплаченных фирмой-потребителем за товар;
- остаток невыплаченных денежных средств (долг фирмы-потребителя).

## **Формирование отчета по фирмам-потребителям**

- a) #Данные# - [Сортировка и фильтр] - [Сортировка] – [Сортировать по] - [Наименование организации];
- b) #Данные# – [Структура] - [Промежуточный итог] и далее нужно сделать необходимые установки в окнах ввода панели «Промежуточный итог»:  
 При каждом изменении в: ..... Наименование организации  
 Операция: ..... Сумма  
 Добавить итоги по: ..... Всего, Оплата, Остаток  
 закончить установки щелчком левой кнопки мыши на кнопке «OK».
- c) выделить из всей таблицы **только итоги** щелчком левой кнопки мыши на кнопке 2 уровней структуры.

Табл.2.2.

Наим.орг.	Всего	Оплата	Остаток
АЛЬЯНС Итог	125	0	125
АННА-МАРИЯ Итог	1092,5	0	1092,5
АЯКС Итог	115	0	115
БЕЛЬКАНТО Итог	78	0	78
ВИЛАРД Итог	352,5	100	252,5
ВОЛАНД Итог	46	0	46
КАО+ Итог	735	0	735
КАРАТ Итог	567	0	567
КОНДАС Итог	1668	25	1643
ЛИГА Итог	410	0	410
ЛОРА Итог	418	200	218
НЕСТЕХ Итог	120	0	120
ОЛФОЮ Итог	122,5	0	122,5
ПЕТРО-ПАРМА Итог	245	0	245
СОЛО Итог	115	0	115
СТО Итог	122,5	0	122,5
ТИГ Итог	355	120	235

В результате можно получить более наглядный для анализа документ (Табл. 2.2). Столбцы, не содержащие данных, скрыты. Для скрытия столбцов нужно пометить столбцы, затем вызвать контекстное меню и выбрать пункт «Скрыть».

## **Формирование отчета по датам**

- a) #Данные# - [Сортировка и фильтр] - [Сортировка] – [Сортировать по] - [Дата];
- b) #Данные# – [Структура] - [Промежуточные итоги] и далее нужно сделать необходимые установки в окнах ввода панели «Промежуточные итоги»:  
 При каждом изменении в: .....Дата  
 Операция: .....Сумма  
 Добавить итоги по: .....Всего, Оплата  
 затем щелкнуть левой кнопкой мыши на кнопке [OK];

- c) выделить из всей таблицы **только итоги** щелчком левой кнопки мыши на кнопке 2 уровней структуры.

### Удаление итогов

- a) пометить таблицу;  
 b) #Данные# – [Структура] - [Промежуточный итог] - [Убрать все].

### 2.10. Работа со сводными таблицами

Преобразование данных из обычной таблицы Excel 2016 в сводную позволяет представить данные в более наглядной форме за счет группировки данных и промежуточных итогов.

Для создания сводной таблицы нужно:

- a) пометить исходную таблицу;  
 b) #Вставка# - !Таблицы! - [Сводная таблица];  
 c) задать параметры группировки по строкам и столбцам сводной таблицы, содержимое поля реквизитов (поле данных) путем буксировки мышью соответствующих табличек с названиями полей исходной таблиц в шапку, боковик и поле реквизитов классического макета сводной таблицы (Рис. 2.13 ) или в области

Рис. 2.13

«Столбцы», «Строки», «Значения» на всплывающей панели «Поля сводной таблицы». Пример размещения полей значений приведен на рис. 2.14.

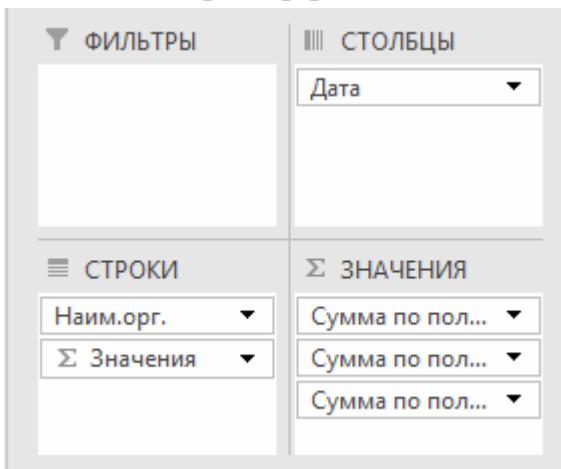


Рис. 2.14.

Для настройки параметров полей значений в области «Значения» открыть поле со списком, выбрать пункт «Параметры полей значений...» и выбрать вид операции: Сумма, Количество, Среднее и т.д. (см. рис. 2.15).

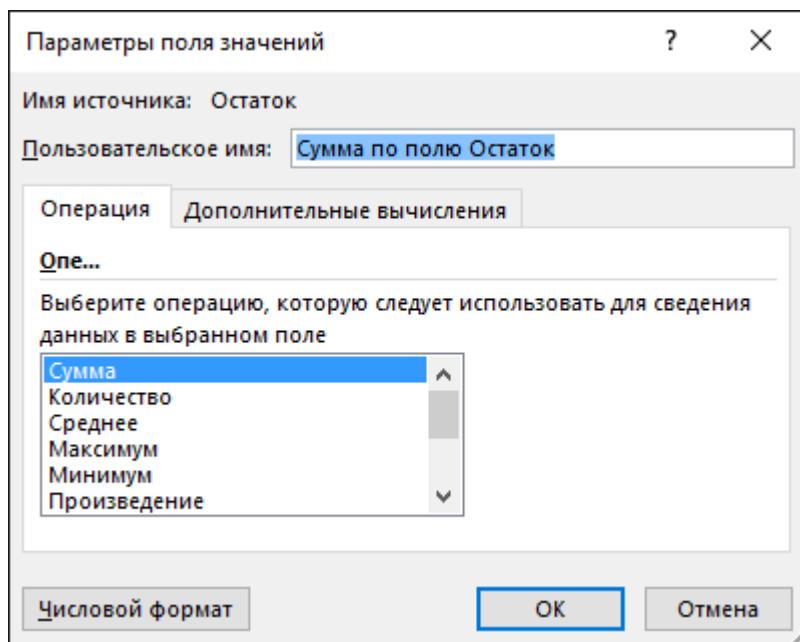


Рис. 2.15

## 2.11. Решение задач

### Подбор параметра

Применим к таблицам, содержащим один столбец данных (такой тип таблицы называют анкетой). Пример такой таблицы приведен на рис. .

Одна из ячеек, значение в которой вычисляется по формуле, объявляется целевой, желаемое значение в ней задается пользователем. (В приведенном примере целевой ячейкой может быть ячейка В14 – Прибыль от продукции) Пользователь должен указать также изменяемую ячейку, содержащую числовое значение (не формулу!), подбирая которое требуется достичь желаемое значение в целевой ячейке. В данном примере такой ячейкой может быть, например, В18 – Себестоимость продукции. В этой таблице приведена калькуляция расходов и доходов от издания книги. Прибыль от издания книги зависит от тиража, % накладных расходов, цены и себестоимости.

Табл. 2.3.

	A	B
1	Расходы/доходы от издания книги	
2		
3		
4	Количество экземпляров	20 000
5	Доход	=B17*B4
6	Себестоимость	=B18*B4
7	Валовая прибыль	=B5-B6
8	% накладных расходов	30
9	Затраты на зарплату	=250*B4
10	Затраты на рекламу	=50*B4
11	Накладные расходы	=B5*B8/100
12	Валовые издержки	=B9+B10+B11
13		
14	Прибыль от продукции	=B7-B12
15		
16		
17	Цена продукции	6 000
18	Себестоимость продукции	2 000
19		

Последовательность действий:

- пометить целевую ячейку в таблице;
- #Данные# - !Прогноз! – [Анализ «что-если»] – [Подбор параметра];

- c) во всплывающем диалоговом окне в поле «Значение» ввести желаемое значение в целевой ячейке и в поле «Изменяя значение ячейки» - адрес изменяемой ячейки

- в окно «Установить в ячейке:»- адрес целевой ячейки;
- в окно «Значение:» - желаемое значение в целевой ячейке;
- в окно «Изменяя значение ячейки:» - адрес изменяемой ячейки;

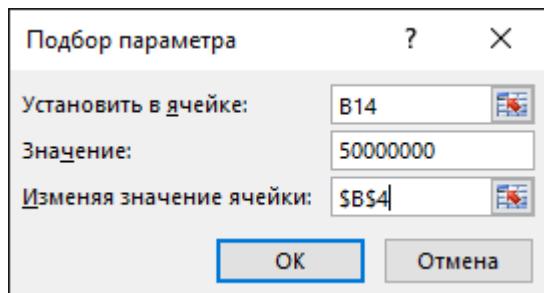


Рис. 2.16

### Поиск оптимальных решений

Применим к любым таблицам Excel 2016. Позволяет решать задачи оптимизации при наличии ограничений, а именно: найти в условиях ограничений на допустимые значения чисел такую комбинацию числовых значений в изменяемых ячейках одной или нескольких строк в таблице, при которой значение в целевой ячейке будет максимальным, минимальным или равным нулю. В качестве примера ниже приведена таблица (Табл. 2.4) с калькуляцией доходов и расходов предприятия.

Задача оптимизации может заключаться в том, чтобы найти условия получения максимальной годовой прибыли при тех же затратах, но при другом распределении затрат (например, на рекламу) по кварталам.

Для решения задачи оптимизации при наличии ограничений в Excel 2016 предусмотрена программа Solver. После стандартной установки Excel 2016 программа Solver не устанавливается. Для ее установки:

#Файл# – [Параметры] – [Надстройки] - [Поиск решения] – [Перейти].

Табл. 2.4.

<b>Калькуляция расходов и доходов предприятия</b>					
<b>Квартал</b>	<b>I</b>	<b>II</b>	<b>III</b>	<b>IV</b>	<b>Всего</b>
<b>Сезонный фактор</b>	0,9	1,1	0,8	1,2	
<b>Объем сбыта</b>	3592	4390	3192	4789	15962
<b>Доход от оборота</b>	143662	175587	127700	191549	638498
<b>Себестоимость реализованной продукции</b>	89789	109742	79812	119718	399061
<b>Валовая прибыль</b>	53873	65845	47887	71831	239437
					0
<b>Затраты на зарплату</b>	8000	8000	8000	8000	32000
<b>Затраты на рекламу</b>	10000	10000	10000	10000	40000
<b>Накладные расходы</b>	21549	26338	19155	28732	95775
<b>Валовые издержки</b>	39549	44338	37155	46732	167775
					0
<b>Прибыль от продукции</b>	14324	21507	10732	25099	71662
<b>Коэффициент прибыльности</b>	10%	12%	8%	13%	11%
<b>Цена продукции</b>	40				
<b>Себестоимость продукции</b>	25				

В результате на вкладке «Данные» появится группа «Анализ» с пиктограммой «Поиск решения».

Последовательность действий при решении задачи оптимизации:

- пометить целевую ячейку;
- #Данные# - !Анализ! – [Поиск решения].

В результате появится диалоговое окно «Параметры поиска решения» (рис. 2.15), в котором:

- в окно «Оптимизировать целевую функцию:» ввести адрес целевой ячейки:

- активизировать один из элементов (Максимум, Минимум) в группе «До:» или задать фиксированное значение, достигаемое в целевой ячейке в результате оптимизации в окно «Значения»;

- указать диапазон изменяемых ячеек в окне «Изменяя ячейки переменных:»;
- ввести ограничения в окно «В соответствии с ограничениями». Вызов панели ввода ограничений - щелчком левой кнопки мыши на кнопке «Добавить».

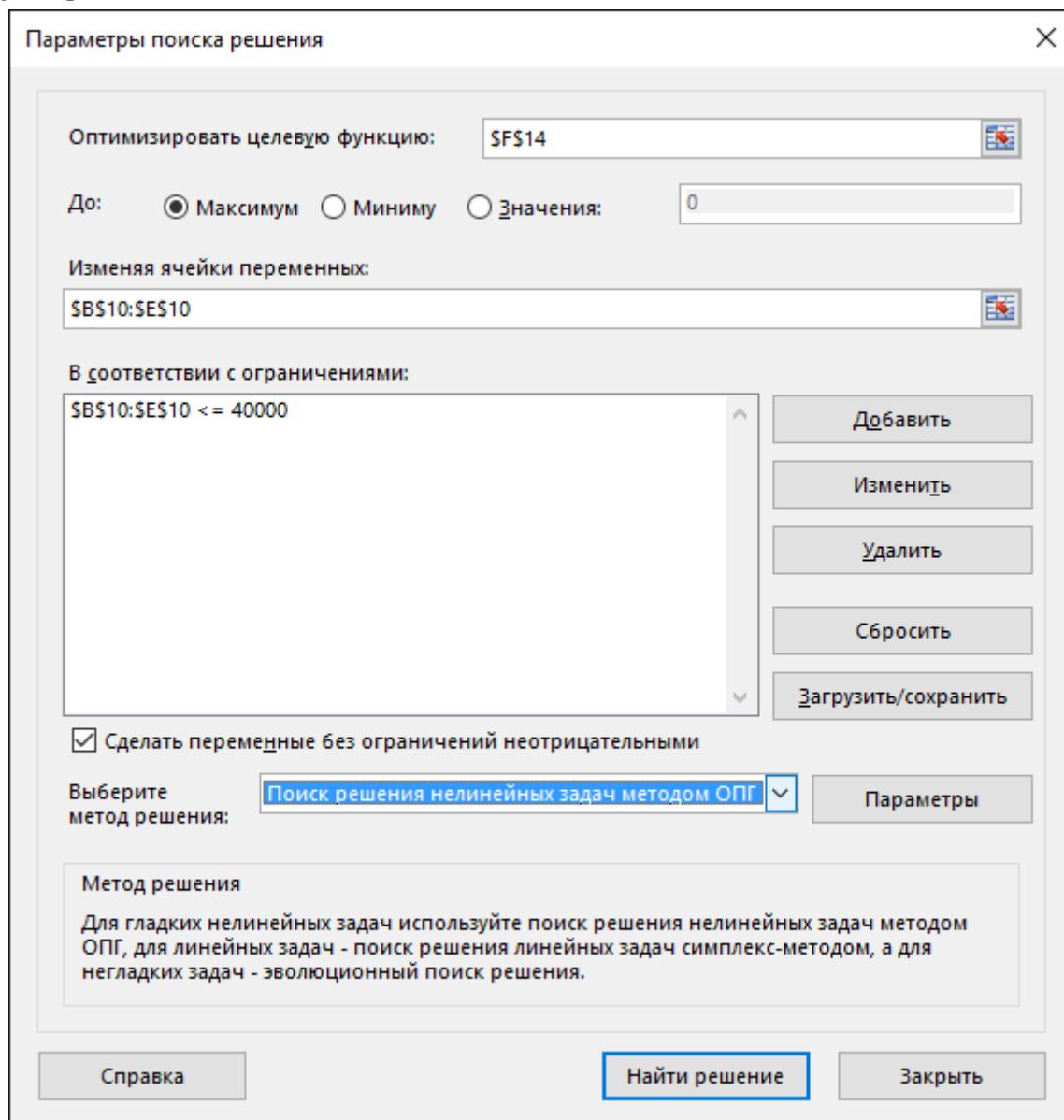


Рис. 2.15.

Для получения автоотчета по результатам оптимизации во всплывающей после активизации кнопки «Найти решение» панели «Результаты поиска решения» нужно выбрать: «Отчеты/Результаты» (см. рис. 2.16).

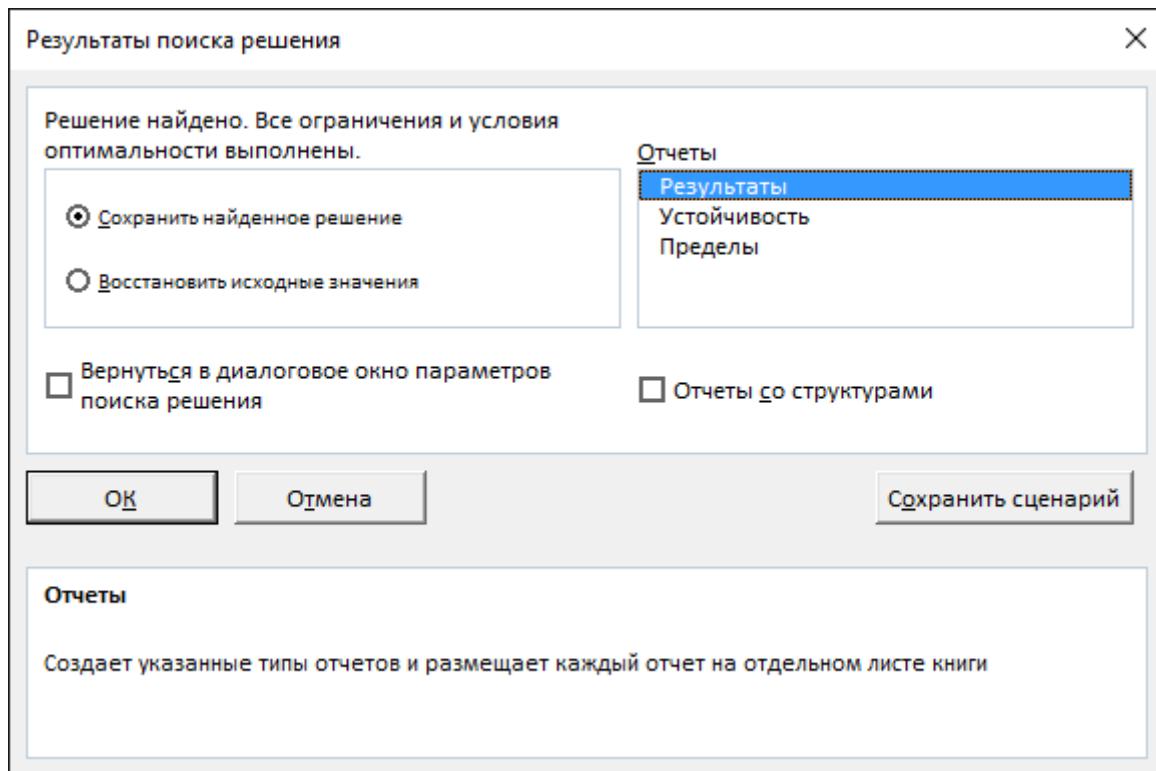


Рис. 2.16

## 2.12. Создание документов на основе данных в таблицах

Средствами Excel 2016 можно формировать документы типа счета, накладной и др. на основе данных, содержащихся в таблицах. Типичным примером является формирование накладной по записи в базе данных, в которой фиксируются результаты продаж. Каждой продаже в этой таблице соответствует одна строка таблицы. Для извлечения данных из любой строки такой таблицы можно использовать функцию ВПР (VLOOKUP) из категории «Ссылки и массивы».

Параметрами этой функции являются:

Искомое\_значение – значение, находящееся в первом столбце в выбираемой строке таблицы (например, в первом столбце можно разместить номер записи или номер квитанции);

Таблица – диапазон ячеек таблицы;

Номер столбца – номер столбца таблицы, из которого извлекаются данные. (Номер строки, из которой извлекаются данные, уже определен первым параметром). На рис. 2.17 приведен вид всплывающей панели для ввода параметров функции «ВПР».

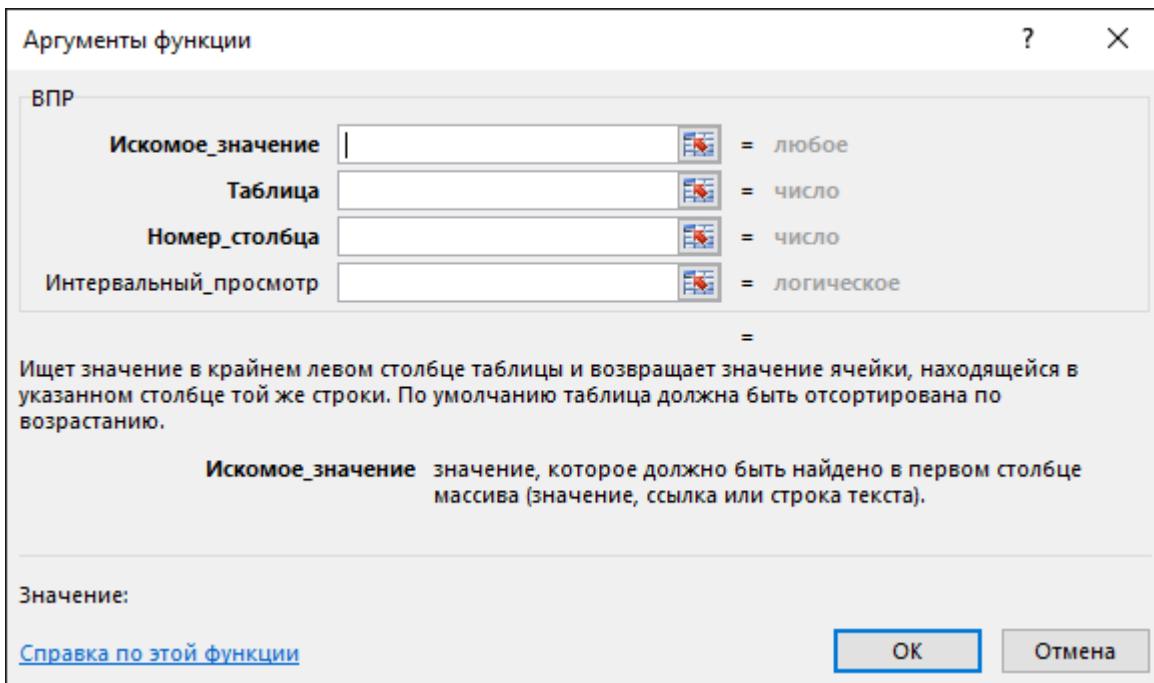


Рис. 2.17.

### **2.13. Работа с макросами**

В макросы записывают последовательности команд, соответствующих выполнению нескольких часто повторяющихся последовательных действий с таблицами или диаграммами Excel 2016. Запуск макроса будет приводить к автоматическому выполнению всей последовательности действий. Для возможности работы с макросами нужно установить низкий уровень защиты:

#Файл# – [Параметры] – [Центр управления безопасностью] – [Параметры центра управления безопасностью] – [Параметры макросов] – [Включить все макросы].

Удобные средства для работы с макросами представлены на вкладке РАЗРАБОТЧИК. После стандартной установки Excel 2016 эта вкладка отсутствует. Для ее установки:

- a) #Файл# – [Параметры] ;
- b) во всплывающем окне «Параметры» выбрать пункт «Основные»;
- c) установить флагок «Показывать вкладку РАЗРАБОТЧИК на ленте»

## **Создание макроса**

- a) #Разработчик# - !Код! - [ Запись макроса];
- b) ввести имя макроса в окно ввода “Имя макроса” панели “Запись макроса”;
- c) выполнить действия с таблицей или диаграммой;
- d) остановить запись макроса: #Разработчик# - !Код! - [ Остановить запись].

*Примечание: макрос будет создаваться на отдельном листе в открытой рабочей книге. Можно создать отдельную рабочую книгу специально для макросов и начать создание макроса, находясь в ней.*

## **Запуск макроса**

- a) #Вид# - !Макросы! - [Макросы...];
- b) выбрать из меню нужный макрос;
- c) [Выполнить].

## **Создание средств запуска макросов**

Для запуска макросов могут быть созданы и использоваться:

- a) кнопки, которые могут быть добавлены на рабочий лист Excel;
- b) графические объекты и объекты WordArt.

Для создания кнопки на рабочем листе Excel нужно выполнить следующие действия:

- a) #Разработчик# – !Элементы управления! – [Вставить];
- b) выбрать объект “Кнопка” и установить его в нужном месте листа рабочей книги Excel и нужного размера;
- c) связать кнопку “Кнопка” с макросом для чего во всплывающем диалоговом окне “Назначить макрос объекту” указать требуемый макрос в списке.

В результате на рабочем листе появится кнопка, с помощью которой можно будут запускать макросы. Надпись на кнопке можно изменить, вызвав в контекстном меню пункт «Изменить текст».

Для создания средства запуска макроса в форме графического объекта или объекта WordArt нужно создать такой объект и связать его с макросом так же, как описано выше: т.е. вызвать контекстное меню правой кнопкой мыши и выбрать пункт “Назначить макрос объекту”, затем указать требуемый макрос в списке.

## **2.14. Выход из Excel 2016**

Для выхода из Excel 2016 в среду Windows нужно закрыть все открытые файлы:

#Файл# - [Закрыть].

### **Приложение 1. Клавиши быстрого доступа**

Клавиша	Действие
Ctrl+O	Открыть файл
F12	Сохранить файл как...
Ctrl+S	Сохранить файл
Ctrl+Z	Отменить действие
Ctrl+Y	Повторить действие
Alt+=	Суммирование
Shift+F3	Вставка функции
Ctrl+F3	Диспетчер имен
Ctrl+C	Копировать в буфер хранения
Ctrl+X	Перемесить в буфер хранения (Вырезать)
Ctrl+V	Вставить из буфера хранения
Shift+F11	Вставить лист
Ctrl+Alt+Tab	Увеличить отступ
Ctrl+Alt+Shift+Tab	Уменьшить отступ
Ctrl+%	Процентный формат
Shift+F9	Автоматический пересчет формул в книге
F9	Автоматический пересчет формул на листе
Ctrl+Alt+L	Повторная сортировка
Alt+F5	Обновить источник данных

**Приложение 2. Клавиши быстрого перемещения по рабочему листу и книге.**

Клавиши	Действие
← , ↑, →, ↓	Перемещение на одну ячейку влево, вверх, вправо и вниз соответственно
Page Up, Page Down	Перемещение на одну страницу вверх или вниз соответственно
Ctrl+Page Up, Ctrl+Page Down	Перемещение к предыдущему листу, к следующему листу соответственно
Ctrl+Home, Ctrl+End	Перемещение в ячейку A1, в ячейку правого нижнего угла таблицы
Tab	Перемещение к следующей ячейке вправо
Enter	Перемещение к следующей ячейке вниз
Shift+Tab	Перемещение к следующей ячейке влево
Shift+Enter	Перемещение к следующей ячейке вверх
Alt+Page Up	Перемещение к следующему экрану влево
Alt+Page Down	Перемещение к следующему экрану вправо
Ctrl+Backspace	Возврат к активной ячейке, которая вышла за пределы экрана в результате прокрутки листа
End+←	Переход в первый столбец таблицы или пустого листа
End+↑	Переход в первую строку таблицы или пустого листа
End+→	Переход в последний столбец таблицы или пустого листа
End+↓	Переход в последнюю строку таблицы или пустого листа

## 2.15. Упражнения

Подготовка, редактирование и печать таблиц

1. Элементарные действия при создании новой таблицы.

Создайте следующую таблицу:

Табл. 2.5.

	Затраты в себестоимости	Наименование изделия (пломбы)				
		Фликлок	Унисил	НП	Нейлсил	Трос
1	Сырье и материалы	1,55	1,07	1,14	21,27	28,75
2	Топливо	0,34	0,24	0,26	2,35	3,39
3	Энергия	0,89	0,62	0,69	0,16	0,23
4	Расходы на оплату труда	0,05	0,03	0,04	0,35	0,51
5	Начисления на зарплату	0,01	0,01	0,01	0,07	0,10
6	Административные расходы	0,35	0,24	0,27	0,42	0,61
7	Выплаты, включаемые в себестоимость	0,18	0,12	0,14	1,25	1,80
8	Сбытовые расходы	0,54	0,37	0,42	3,74	5,40
9	Реклама	0,04	0,03	0,03	0,28	0,41
10	Операционные затраты					
11	Налоги в себестоимости	1,26	0,87	0,97	6,72	9,71
12	Амортизация оборуд.	0,04	0,03	0,03	0,06	0,09
13	<b>Себестоимость</b>	.				
14	Процентная прибыль	0,05	0,05	0,06	0,04	0,09
15	Продажная цена без НДС	.				
16	<b>Продажная цена с учетом НДС</b>					

Указания по выполнению

Строку «Операционные затраты» и «Себестоимость» заполняйте с использованием автосуммирования строк 1-9 и 10-12 соответственно; строки «Продажная цена без НДС» и «Продажная цена с НДС» - с использованием вычислений по формулам. Размер НДС считать равным 18%.

2. Редактирование таблицы.

- а) введите денежный стиль во всем числовом поле таблицы;
- б) уменьшите разрядность данных (установить данные в таблице в руб. без указания коп.) ;
- в) произведите автоподбор ширины столбцов таблицы;

г) произведите автоформатирование таблицы в целом (исключая заголовок).

3. Создание и редактирование диаграмм.

а) постройте диаграмму типа линейчатая гистограмма по данным строки «Себестоимость» и «Продажная цена» (функции), названия пломбы (аргументы).

б) добавьте название диаграммы, подписи по осям, координатную сетку, используя средства пиктографического меню вкладки «Формат», всплывающей после формирования диаграммы на рабочем листе Excel.

в) постройте круговую диаграмму по данным строки «Себестоимость».

4. Сохранение рабочей книги Excel 2016 с защитой от несанкционированного доступа.

Сохраните полученный табличный документ в файле tabl\_1.xlsx с паролем защиты от несанкционированного доступа.

### **Создание баз данных и работа с ними**

Создайте базу данных по следующему образцу или загрузите из файла example\_1.xlsx:

Табл. 2.6.

Фирма-производитель	Процессор	f, GHz	RAM, Gb	HDD, Gb	Video, Gb	OS	Цена, руб
Dell Inspiron	AMD	1,8	4	500	2	W8, 64	16290
Lenovo	AMD	2	6	1000	2	W8, 64	18690
Lenovo	Intel	1,7	4	500	1	W8, 32	16990
Asus	Intel	1,7	6	500	2	W8	23650
Lenovo	AMD	1,8	4	500	SMA	W8, 32	14390
PB	Intel	1,8	4	320	1	Linux	12750
Lenovo	Celeron	1,7	2	320	SMA	Free DOS	10290
HP	Intel	2,5	8	1000	2	W8, 64	39990
HP	AMD	1,8	4	500	1	W8, 64	17990
Apple	AMD	2,2	16	256 SSD	SMA	Mac OS	87990
Asus	Intel	2,4	6	1000	2	W8, 64	43990
Asus	Intel	2,4	8	1000	4	W8, 64	42990

1. Сортировка.

Произведите сортировку данных в столбцах "f, MHz", "Цена, \$".

2. Поиск в базе данных.

Произведите автоматический поиск данных в базе данных по следующему критерию: Процессор - Intel или AMD, f, GHz >=1,8GHz, RAM >=4Gb, HDD>=500Gb, Цена, <=20000руб.

### **Анализ данных**

Ведите или загрузите из файла Склад2.xlsx следующую таблицу:

Табл. 2.7.

<u>Да-та</u>	<u>Наим. организ.</u>	<u>Кол. 0,5</u>	<u>Цена</u>	<u>Сумма</u>	<u>Кол. 1,0</u>	<u>Цена</u>	<u>Сумма</u>	<u>Всего</u>	<u>Дата</u>	<u>Оплата</u>	<u>Оста- ток</u>
01.янв	ВИЛАРД	50	2,45	1470			0	1470	15. янв	1400	70
01.янв	КАРАТ	80	2,4	2304			0	2304			2304
01.янв	КОНДАС	50	2,45	1470	20	4,4	79,2	1549,2	05. янв	1000	549,2
01.янв	ЛИГА	20	2,5	600			0	600			600
02.янв	ЛОРА	100	2,6	3120	10	4,3	38,7	3158,7	07. янв	3000	158,7
02.янв	СОЛО	50	2,3	1380			0	1380			1380
02.янв	ТИГР	50	2,5	1500			0	1500			1500
03.янв	АЛЬЯНС	50	2,5	1500			0	1500			1500
03.янв	АННА- МАРИЯ	200	2,4	5760			0	5760			5760
03.янв	АЯКС	50	2,3	1380			0	1380			1380
03.янв	ВИЛАРД	100	2,3	2760			0	2760			2760
03.янв	КОНДАС	100	2,5	3000			0	3000			3000
03.янв	СТО	50	2,5	1500			0	1500			1500
03.янв	ТИГ	100	2,45	2940			0	2940	10. янв	2900	40
04.янв	ОЛФОЮ	50	2,3	1380			0	1380			1380
05.янв	БЕЛЬКАНТ	30	2,45	882			0	882			882
05.янв	ВОЛАНД	20	2,6	624			0	624			624
05.янв	КАРАТ	150	2,3	4140			0	4140			4140
06.янв	ЛИГА	100	2,5	3000			0	3000			3000

1.Формирование отчета по фирмам-потребителям.

Произведите формирование отчета по фирмам.

2.Формирование отчета по датам.

Сформируйте отчет за 1-5 января.

## Сводные таблицы

Создайте по данным предыдущей таблицы (файл Склад2.xlsx) сводную таблицу, в которой будут представлены итоги как по датам, так и по фирмам.

Табл. 2.8.

		Дата												
Наим.орг.	Данные	01. янв	02. янв	03. янв	04. янв	05. янв	06. янв	07. янв	08. янв	09. янв	10. янв		Общий итог	
АЛЬЯНС	Сумма по полю Всего				125								125	
	Сумма по полю Оплата													
	Сумма по полю Остаток				125								125	
АННА -МАРИЯ	Сумма по полю Всего				1092,5								1092,5	
	Сумма по полю Оплата													
	Сумма по полю Остаток				1092,5								1092,5	
ВОЛАНД	Сумма по полю Всего					46							46	
	Сумма по полю Оплата													
	Сумма по полю Остаток					46							46	
Итог Сумма по полю Всего		0	0	1217,5	0	46	0	0	0	0	0		1263,5	
Итог Сумма по полю Оплата		0	0	0	0	0	0	0	0	0	0		0	
Итог Сумма по полю Остаток		0	0	1217,5	0	46	0	0	0	0	0		1263,5	

## Решение задач

### 1. Подбор параметра

а) создайте следующую таблицу:

Табл.2.9.

	A	B
1	Расходы/доходы от выпуска продукции	
2		
3		
4	Количество экземпляров	20 000
5	Доход	=B17*B4
6	Себестоимость	=B18*B4
7	Валовая прибыль	=B5-B6
8	% накладных расходов	30
9	Затраты на зарплату	=250*B4
10	Затраты на рекламу	=50*B4
11	Накладные расходы	=B5*B8/100
12	Валовые издержки	=B9+B10+B11
13		
14	Прибыль от продукции	=B7-B12
15		
16		
17	Цена продукции	6 000
18	Себестоимость продукции	2 000

б) произведите подбор параметра в ячейке B14, выясните условия получения прибыли 50000000 за счет всех возможных изменений факторов по отдельности: количества экземпляров, накладных расходов, цены и себестоимости.

## 2. Поиск оптимальных решений.

а) создайте следующую таблицу:

Табл. 2.10.

	A	B	C	D	E	F
1	Квартал	I	II	III	IV	Всего
2	Сезонный фактор	0,9	1,1	0,8	1,2	
3						
4	Объем сбыта	3592	4390	3192	4789	15962
5	Доход от оборота	143662	175587	127700	191549	638498
6	Себестоимость реализованной продукции	89789	109742	79812	119718	399061
7	Валовая прибыль	53873	65845	47887	71831	239437
8						0
9	Затраты на зарплату	8000	8000	8000	8000	32000
10	Затраты на рекламу	10000	10000	10000	10000	40000
11	Накладные расходы	21549	26338	19155	28732	95775
12	Валовые издержки	39549	44338	37155	46732	167775
13						0
14	Прибыль от продукции	14324	21507	10732	25099	71662
15	Коэффициент прибыльности	10%	12%	8%	13%	11%
16						
17	Цена продукции	40				
18	Себестоимость продукции	25				
	<b>Параметры заполнения</b>					
	B4 = 35*b2*(B10+3000)^0.5					
	B5 = B4*\$B\$17					
	B6 = B4*\$B\$18					
	B7 = B5-B6					
	B11 = 0.15*B5					
	B12 = СУММ(B9:B11)					
	B14 = B7-B12					
	B15 = B14/B5					

б) найдите условия оптимального распределения затрат на рекламу по кварталам, при котором годовая прибыль максимальна.

## Использование функций

Используя функции ТЕНДЕНЦИЯ и РОСТ выполните линейную и экспоненциальную экстраполяцию в приведенной ниже таблице; с помощью экстраполяции произведите прогнозирование экспорта товаров в будущем (например, в 1995г.) по данным за 1990-1994г.г.

Табл. 2.11.

<b>Структура экспорта товаров из России в страны дальнего зарубежья (Российский статистический ежегодник, 1995г.)</b> (в млрд. долларов США)					
	1990	1991	1992	1993	1994
Машины, оборудование и транспортные средства	12.5	5.2	3.8	2.9	2.5
Минеральные продукты (в т. ч. нефть и газ)	32.3	26.3	22	20.7	21.9
Металлы, драгоценные камни и изделия из них	9.2	7.3	7	10.3	13.1
Продукция химической промышленности	3.3	3.4	2.6	2.6	3.9
Древесина и целлюлозно-бумажные изделия	3.1	2.4	1.6	1.9	2.1
Текстиль и текстильные изделия	0.7	0.5	0.3	0.2	0.4
Кожевенное сырье, пушнина и изделия из них	0.1	0.1	0.1	0.1	0.2
Продовольственные товары и сельхоз сырье	1.5	1.3	1.6	1.6	2.1
Прочие	8.4	4.4	3.4	4	3.9
<b>ВСЕГО</b>	<b>71.1</b>	<b>50.9</b>	<b>42.4</b>	<b>44.3</b>	<b>50.1</b>

## **Работа с макросами**

### ***Создание макроса***

- а) откройте файл wares.xlsx;
- б) выполните копирование листа рабочей книги, содержащего базу данных, на Лист 2 этой же книги и одновременно создайте макрос для этой операции.

### ***Запуск макроса***

Выполните копирование листа рабочей книги, содержащего базу данных, на Лист 2 с помощью макроса.

### ***Создание средств управления для работы с макросами***

Создайте средства управления (кнопку управления, графический объект, объект WordArt) для запуска макроса, созданного по п. «Создание макроса».

*Указания по выполнению.*

Для создания кнопки управления  
#Разработчик# - !Вставить! – [Кнопка управления].

### ***Задание 1 для самостоятельной работы***

Необходимо автоматизировать процесс создания ценников на товары, ежедневно поступающие в магазин. Вид ценника - по образцу табл. 2.13. Информация о партии товара содержится в файле wares.xlsx (см. приведенный ниже фрагмент табл. 2.12).

Табл. 2.12.

<b>Код</b>	<b>Наименование</b>	<b>Цена, руб</b>
000000	Вилка	4000,00
010001	УН 5.5/16	15000,00
010002	УН 8.5/25А	12000,00
010003	УН 8.5/25К	8000,00
010004	УН 9/18	10500,00
010005	УН 9/27	11500,00
010006	УН 6/12	3000,00
010008	ВВВ	10000,00
010009	УН 9/27М	9000,00

Табл.2.13.

Россия			010001
		<b>УН 5.5/16</b>	
1997г.	шт.	<b>22000р.</b>	
1998г.	шт.	<b>22р00к</b>	

До формирования ценников необходимо разделить эту БД на две: БД1 и БД2 по признаку длины текстовой строки с наименованием товара, а именно:

БД1 - длина строки <=20;

БД2 - длина строки >20.

В БД2 нужно изменить формат ячеек в столбце с наименованием товара так, чтобы в ценнике наименование товара занимало не одну, а две строки, было центрировано по вертикали и горизонтали.

Требуется создать один или несколько макросов и кнопки управления этими макросами, которые обеспечивают автоматическое выполнение процесса создания ценников двух категорий (с "короткими" и "длинными" названиями) по информации, содержащейся в БД "wares.xlsx".

## *Указания по выполнению*

1. Для определения длины текстовой строки в столбце "Наименование товара" используйте функцию **ДЛСТР(<адрес ячейки>)**.

2. Для разделения исходной БД на две: БД1 и БД2 по признаку длины строки:

#Данные# – !Фильтр! – [Числовые фильтры] , далее задайте условия фильтрации по столбцу OKDP и скопируйте результаты фильтрации на свободные листы в рабочей книге.

3. Для установки переноса по словам и центрирования в клетках столбца "Name" базы БД2 с "длинными названиями": #Главная# – !Ячейки! – [Формат] – [Формат ячеек] – [Выравнивание], далее задать параметры выравнивания и отображения.

4. Оптимальную ширину столбца "Name" и высоту строк в БД2 установите путем автоподбора: #Главная# – !Ячейки! – [Формат] – [Автоподбор ширины столбца/Автоподбор высоты строк].

5. Перед созданием макросов для разделения исходной БД на две: БД1 и БД2 по признаку длины строки выполните процесс создания ценников, не включая запись макроса, чтобы убедиться в правильности всех действий, затем очистите листы с БД1 и БД2, включите запись макроса и повторите все необходимые действия.

6. Для создания ценников по данным, содержащимся в БД1 и БД2, создайте в ручном режиме строку ценников. Количество ценников в строке должно быть равно числу строк в ценнике. (Попробуйте объяснить - почему) Затем пометьте строку ценников и, используя прием автозаполнения, создайте ценники для всех товаров, данные о которых находятся в базе данных.

Для заполнения клеток первого ценника, содержащих данные из БД1 и БД2 используйте прием внесения адресов ячеек БД1 и БД2 в формулы;

При форматировании строки ценников используйте команду "Формат по образцу", а именно: выполните форматирование первого ценника (размер и вид шрифта, центрирование, денежный стиль, рамку), затем скопируйте формат группы столбцов, в которых находится первый ценник, на группу столбцов, в которых находится остальные ценники.

## Задание 2 для самостоятельной работы

Автоматизируйте формирование расчетных листков сотрудников на основании таблицы расчета зарплаты, приведенной в файле Зарплата.xlsx. Сокращенный пример таблицы расчета зарплаты приведен ниже.

Табл. 2.14

### Расчетная ведомость по зарплате сотрудников

№	ФИО сотрудника	Подразделение	Отработано дней	Оклад	Премия	Отпуск	Больничный	Итого начислено
1	Кожевкина Н.Ф.	ОГМ	20	60801,1	899	2141,56	0	63841,66
2	Кузнецов А.П.	ОГМ	21	40037,78	20		0	40057,78
3	Нежная Л.В.	КО	21	40037,78	0		0	40037,78
4	Сидоров П.Е.	ТО	21	20018,75	0		0	20018,75
5	Васечкин В.А.	ТО	12	10548,8	2147,57		0	16027,37

### Указания по выполнению

Позиции таблицы расчета зарплаты, выделенные жирным шрифтом, вносятся в макет расчетного листка сотрудника и располагаются по вертикали (см. Табл. 2.15).

Позиции образца таблицы, записанные обычным шрифтом, должны заполняться данными из помеченной пользователем строки таблицы расчета зарплаты. Для переноса данных в расчетный листок сотрудника используйте функцию ВПР. Для автоматического формирования расчетного листка по данным в выбранной строке базы данных создайте макрос и кнопку запуска макроса.

Табл. 2.15

**Расчетный лист за февраль 2014**

	ФИО	Юдин Анатолий Сергеевич
	Должность	Инженер 1 кат.
	Отработанное время (дни/ часы)	0
	ОКЛАД, ОТРАБ. руб.	25 / 150
	коэф. к окл.	8410
	компенсация к окладу	2
Надбавки	за раб.со свед. сост. ГТ	8410
	за стаж непр. раб. с ХО	0
	Оплата по среднему заработка руб.	10615.62
	Доплата за совме- щение	0
	Премия	0
Доплаты	заочные часы работы	1720
	за раб. в праздн. дни	2018.44
Б/лист	за счет работодателя	672.8
	за счет ФСС	0
Отпуск	учебный	0
	очередной	0
	за свой счет	0
	Компенсация отпуска при увольн.	0
	ИТОГО за месяц, руб.	0
	НДФЛ	31846.86
	По исполнит. листву	0
	ИТОГО за месяц к получению, руб.	31846.86

### 3. Visual Basic for Applications

Visual Basic for Applications (VBA) относится к языкам объектно-ориентированного программирования. Приложения пользователя на языке VBA могут создаваться для работы в среде Word, Excel, Access.

#### 3.1. Основные средства VBA в Excel 2016

**ОБЪЕКТ.** Библиотека встроенных объектов VBA в среде Excel 2016 содержит более 100 объектов, находящихся на различных уровнях иерархии. Примеры объектов: Application (Приложение), Workbooks (Рабочая книга), Worksheets (Лист рабочей книги), Range (Диапазон ячеек), Chart (диаграмма), CommandBar (Панель инструментов). При ссылке на объект нужно указывать не только имя объекта, но и "путь к нему", например:

```
Application.Workbooks("WARES").Worksheets("База1").Range("A1:A12")
```

Если нужно запрограммировать подряд несколько действий, то "путь к объекту" можно указать один раз, используя конструкцию With - End With, например:

```
With ActiveSheet
```

```
    .Range("B6").Value=17
```

```
    .Range("B7").Value=100
```

```
End With
```

**МЕТОД** определяет действие (например, Clear, Copy, Cut, Delete), которое будет совершаться над объектом. Синтаксис применения метода:

Объект.Метод(<список аргументов>).

Примеры команд программы, определяющих действия над объектами:

UserForm.Hide - UserForm закрыть;

Range("B20:B25").Activate

Методы объекта Range могут использовать встроенные в Excel 2016 команды: DataSeries, AutoFill, AutoFilter, AdvancedAutoFilter, Consolidate, Find, GoalSeek, Sort, Subtotal и др.

*Пример.*

```
Range("B8").GoalSeek Goal:=p, ChangingCell:=Range("B7")
```

**СВОЙСТВО** представляет атрибут объекта, определяющий его характеристики, например, цвет или размер. Синтаксис команды установки значения свойства:

Объект.Свойство=ЗначениеСвойства. Примеры:

UserForm1.Caption="<заголовок формы>"

Range("B4").Value="Объем ссуды"

Range("B5").Value=k

Range("B5").Value=35000

Range("B5").Formula="=A1\*B17"

Data=Application.PV(a,b,c)

Range("B5").NumberFormat="#,##0\$"

Range("B5").NumberFormat="0.00%"

CommandButton1.ControlTipText="Это - кнопка отмены"

CommandButton1.Cancel=True

Основные свойства объекта Range, используемые при программном создании таблиц Excel 2016: Value, Formula, FormulaLocal.

Range("B3").Value="Число выплат"

iMarg=Range("B20").Value

Range("B5").Formula="=PV(B7,B2,-B4)"

Range("B5").FormulaLocal=" =ПС(B7;B2;-B4)" - в русскоязычной версии

VBA

**СОБЫТИЕ** представляет собой действие (например, открытие UserForm, щелчок мышью, нажатие клавиши), распознаваемое объектом. С событием можно связать процедуру на VBA, которая будет запускаться на исполнение при возникновении события.

Синтаксис заголовка процедуры обработки события:

Private Sub <имя объекта>\_<имя события>

Пример заголовка процедуры обработки события инициализации (открытия) окна UserForm2:

Private Sub UserForm2\_Initialize()

Программа на VBA может содержать ключевые слова, операторы (присваивания, безусловного и условного перехода, выбора варианта, цикла), переменные, константы и выражения.

*Примечание: приведенный выше и в приложении перечень объектов, методов, свойств и событий представляет очень малую часть полного перечня, однако вполне достаточен для того, чтобы познакомиться с тем, как создаются приложения пользователя.*

### 3.2. Применение VBA

VBA может быть использован для создания:

- ◆ модулей;
- ◆ пользовательских форм (UserForm);
- ◆ модулей класса (новых объектов с их методами и свойствами).

### 3.3. Открытие окна для разработки программ на VBA

Для открытия окна для разработки программ на языке Visual Basic for Application необходимо добавление вкладки «Разработчик».

Для этого:

- a) #Файл# – [Параметры] ;
- б) во всплывающем окне «Параметры» выбрать пункт «Основные»;
- с) установить флажок «Показывать вкладку РАЗРАБОТЧИК на ленте»

Для открытия окна Visual Basic for Application или пользовательских форм нужно открыть из окна Excel 2016 окно Visual Basic (рис. 3.1):

- a) #Разработчик# - !Код! – [Visual Basic]

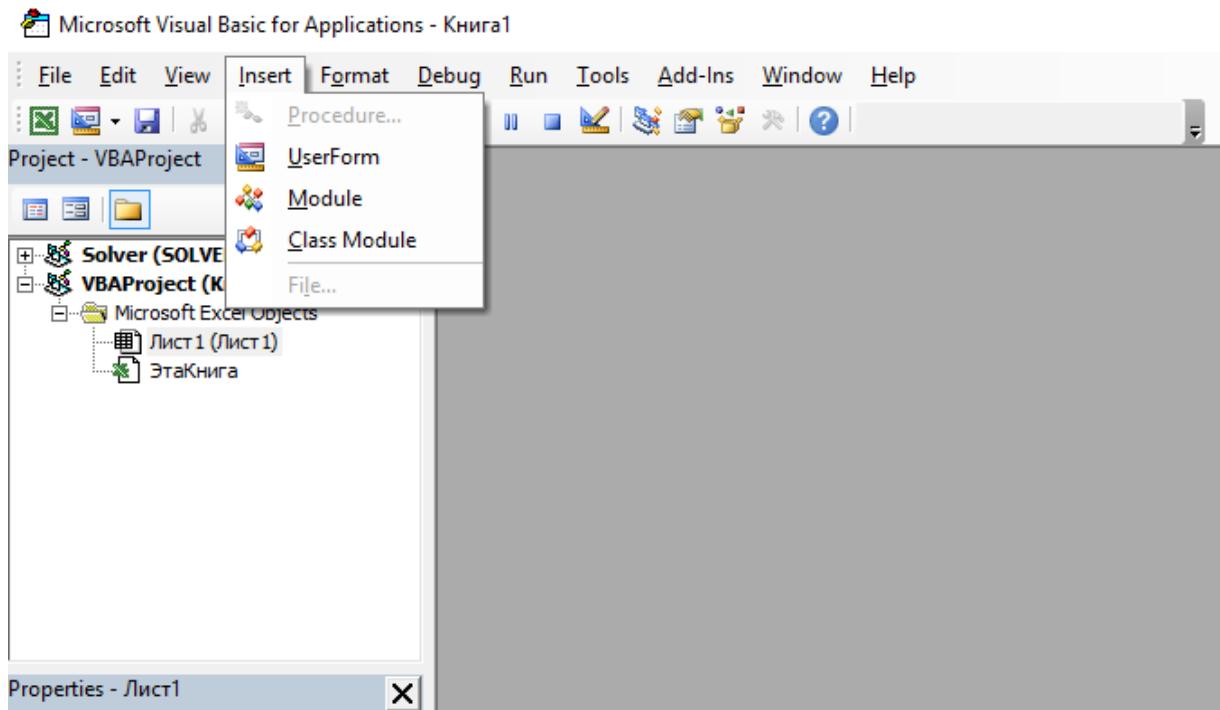


Рис. 3.1.

- b) во всплывающем окне Microsoft Visual Basic открыть окно для создания модуля:

[Insert] - [Module],  
пользовательской формы:  
[Insert] - [UserForm],  
или модуля класса:  
[Insert] - [Class Module].

### **Создание модуля**

Модуль на языке VBA создается в форме определяемой пользователем функции, которая, после ее написания в форме программы, будет автоматически встроена в список функций Excel 2016 .

*Пример модуля.*

```
Function Стоимость(ЦенаОднойКниги, Количество, Скидка)
'Функция пользователя "Стоимость" возвращает значение стоимости
'одной книги с учетом скидок при оптовой покупке (7%, 10% и 15% )
'и скидки постоянному клиенту (5%)
If Количество < 100 Then
    СтоимостьБезСкидки = ЦенаОднойКниги * Количество
Else
    If Количество <= 200 Then
        СтоимостьБезСкидки = ЦенаОднойКниги * Количество * 0.93
    Else
        If Количество <= 300 Then
            СтоимостьБезСкидки = ЦенаОднойКниги * Количество * 0.9
        Else
            СтоимостьБезСкидки = ЦенаОднойКниги * Количество * 0.85
        End If
    End If
End If
If Скидка = 0 Then
    Стоимость = СтоимостьБезСкидки
Else
    Стоимость = СтоимостьБезСкидки * 0.95
End If
End Function
```

*Примечание: апостроф в начале строки – признак комментария.*

## Создание UserForm

Для перехода в режим создания пользовательских форм нужно:

- a) открыть из окна Excel 2016 окно Visual Basic :  
#Разработчик# - !Код! – [Visual Basic];
- b) во всплывающем окне Visual Basic открыть окно для создания пользовательской формы:  
[Insert] - [UserForm].

В результате появится панель UserForm и панель элементов для ее конструирования.

Основными элементами UserForm являются:

**TextBox** - окно для ввода и вывода текста;

**CommandButton** - кнопка, используемая для запуска процедуры на VBA;

**OptionButton** - кнопка, возвращающая значение TRUE, если она нажата, и FALSE - если нет;

**ToggleButton** - выключатель, устанавливает одно из двух состояний (TRUE или FALSE ) или одно из трех состояний (TRUE, FALSE или NULL);

**SpinButton** - счетчик, возвращает текущее значение, при активизации кнопок счетчика число может увеличиваться или уменьшаться;

**CheckBox** - флажок, устанавливает одно из двух состояний (TRUE или FALSE ) или одно из трех состояний (TRUE, FALSE или NULL);

**Label** - возвращает текст, отображаемый в надписи;

**Frame** - рамка, визуально выделяет группы элементов управления в UserForm.

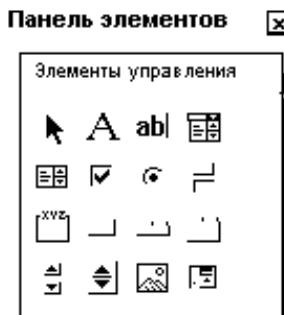
Для элементов управления определены МЕТОДЫ, СВОЙСТВА и СОБЫТИЯ, например:

Text.Box5.Enabled=False - запрет ввода текста в окно TextBox5;

TextBox6.Text=<текст>;

TextBox2.Text=Cstr(Format(p,"Fixed")) - переформатирование значения переменной p из числовой формы в текстовую и вывод в окно TextBox2.

## Панель элементов VBA



- ☒ Для создания средств пользовательского интерфейса VBA предоставляет следующие элементы:
  - Label** - для создания надписей в UserForm , например, заголовка текстового окна;
  - TextBox** - для ввода и вывода значений (строк текста и чисел);
  - Можно использовать TextBox для ввода пароля. Для того, чтобы в процессе ввода пароля вместо вводимых символов отображались, например, звездочки, нужно в окне свойства PasswordChar текстового окна ввести символ \*.
  - Пример анализа пароля в программе:

```
If TextBox1.Text <> "TVS" Then
    Exit Sub
End If
```

  - ComboBox** - для хранения списка значений. Отображается только один элемент списка;
  - ListBox** - для хранения и отображения списка значений. Из списка можно выбрать одно значение, которое будет использовано в программе, запускаемой нажатием командной кнопки;
  - CheckBox** - для ввода одного из двух (True, False) или (если установить True свойства TripleState) одного из трех (True, False, Null) значений;
  - ToggleButton** - выполняет те же функции, что и флажок;
  - OptionButton** - выполняет те же функции, что и флажок, но, если в UserForm или в группе (группировка переключателей производится элементом Frame) переключателей несколько, в состояние True можно установить только один переключатель - остальные автоматически перейдут в состояние False;
  - Frame** - для группировки элементов управления. Основное свойство рамки - Caption, позволяющее задать название группы элементов;
  - CommandButton** - для запуска программы;
  - TabStrip** - создаются в UserForm, элементы управления могут размещаться на вкладках, однако, вкладки не обладают свойствами контейнеров. Это значит, что элементы управления фактически будут связаны с UserForm и будут только "просвечивать" через все вкладки.
  - Можно сделать программным путем элементы управления видимыми на одних вкладках и невидимыми на других.

Пример.

```
Private TabStrip1_Change()
If TabStrip1.Value = 0 Then
    CommandButton1.Visible=False
Else
    CommandButton1.Visible=True
End If
End Sub
```

В результате действия этой процедуры кнопка CommandButton1 будет видна и будет действовать на вкладке 0 и не будет видна и не будет действовать на вкладке 1 и других.

**Pages** - создаются в UserForm, элементы управления могут размещаться на страницах. Страницы обладают свойствами *контейнеров*. Это значит, что элементы управления будут связаны с теми страницами, на которых они установлены, видны и действовать только на них;

**ScrollBar** - возвращает целое неотрицательное число.

Пример. При каждом нажатии кнопок ScrollBar или перемещении ползунка число в счетчике изменяется и выводится в TextBox.

```
Private Sub ScrollBar1_Change()
spin = ScrollBar1.Value
TextBox2.Text = CStr(Format(spin, "Fixed"))
End Sub
```

**SpinButton** - выполняет те же функции, что и **ScrollBar**, но не содержит ползунка.

Пример. При каждом нажатии кнопок SpinButton число в счетчике изменяется и выводится в TextBox

```
Private Sub SpinButton1_Change()
spin = SpinButton1.Value
TextBox2.Text = CStr(Format(spin, "Fixed"))
End Sub
```

**Image** - для "украшения" UserForm рисунком или мозаикой из рисунков (в последнем случае нужно установить True свойство PictureTiling);

**RefEdit** - используется для ввода ссылок на ячейки или диапазоны.

**Для создания UserForm:**

1. Сконструировать панель UserForm, т.е. разместить на этой панели нужные элементы и определить их свойства (вызвать контекстное меню и выбрать пункт “Properties”). Пример внешнего вида панели UserForm приведен на рис. 3.2.

2. Для кнопок “CommandButton” создать запускаемые ими процедуры на языке VBA. Для этого:

а) двойным щелчком на изображении кнопки CommandButton вызвать окно программы. В этом окне автоматически создается первая и последняя строка процедуры:

```
Private Sub CommandButtonN_Click()
End Sub
```



Рис. 3.2.

б) ввести текст процедуры. Ниже приведен пример текста процедуры.

```
Private Sub CommandButton3_Click()
x = CDbl(TextBox2.Text)
Range("B3").Value = x

Range("B4").Formula = "=ACOS(B3)"
y = Range("B4").Value
TextBox1.Text = CStr(Format(y, "#####0.0#####"))
End Sub
```

Дополнительные элементы пользовательского интерфейса, которые могут использоваться в программах на VBA:

- ◆ всплывающие окна с комментариями к элементам управления в UserForm;
- ◆ функции InputBox и MsgBox.

Всплывающие окна с комментариями могут создаваться программно или создаваться в окне свойств элемента.

Функция InputBox выводит на экран диалоговое окно, содержащее сообщение и поле ввода, кнопки "OK" и "Cancel". Устанавливается режим ожидания ввода текста пользователем или нажатие кнопки, затем функция возвращает из текстового окна значение типа String, содержащее текст, введенный в поле.

Синтаксис (возможный):

`InputBox("<Текст>","<Название окна>")`

Пример: Имя = InputBox("Введите Ваше имя", "Первый шаг")

Функция MsgBox выводит на экран диалоговое окно, содержащее сообщение, и одну или несколько кнопок управления (состав кнопок задается программно)

Функция возвращает число формата Integer, соответствующее тому, какая кнопка была нажата. Синтаксис:

`MsgBox("<Текст>,[<имя набора кнопок>],"<имя окна>")`

Варианты имен наборов кнопок:

VbOKOnly, VbOKCancel, VbAbortRetryIgnore, VbYesNoCancel, VbYesNo, VbRetryCancel.

Пример:

`A = MsgBox ("Справку вывести?",vbOKCancel)`

```

Select Case A
Case vbOK
    MsgBox "<текст>[,<значок сообщения>],[<имя окна>]"
End Select
Возможные виды значков сообщения: vbInformation, vbCritical,
vbQuestion, vbExclamation.

```

### ***Отображение диаграмм***

Из программы на VBA можно создавать диаграммы на основе данных в таблицах и отображать созданные диаграммы на панели UserForm в окнах Image. Для отображения диаграммы нужно вначале экспорттировать диаграмму в рисунок в одном из графических форматов: bmp, jpg, gif и сохранить в файле, затем прочитать из файла и перенести в окно Image. Ниже приведен пример программы.

```

Private Sub CommandButton1_Click()
'Пометка строки с данными для графика
Range("A2:K2").Select
'Построение диаграммы. Перенесен текст макроса
ActiveSheet.Shapes.AddChart.Select
ActiveChart.SetSourceData Source:=Range("Лист1!$A$2:$K$2")
ActiveChart.ChartType = xlLineMarkers
'Преобразование диаграммы в изображение формата bmp и запись в файл
ActiveChart.Export Filename:="D:\graph1.bmp", FilterName:="BMP"
'Загрузка изображения из файла в окно Image1
Image1.Picture = LoadPicture("D:\graph1.bmp")
'Установка режима отображения Stretch в окне
Image1.PictureSizeMode = 1
End Sub

```

### ***Запуск приложения из UserForm***

Для запуска проекта (приложения) из окна Visual Basic  
[Run] - [Run Sub/UserForm]

В результате произойдет переход из режима редактирования проекта в режим запуска; окно VBA закроется, откроется окно Excel 2016, в нем появится UserForm, которой можно пользоваться.

Закрывая окно UserForm, мы возвращаемся в режим редактирования.

*Для запуска проекта из окна Excel 2016*

а) [Разработчик] - [Код] - [Visual Basic];

б) [View] - [Project Explorer];

в) выбрать из иерархии проектов нужный, например VBA project:  
Forms/UserForm1

*Для закрытия проекта достаточно закрыть окно проекта, но можно закрыть проект и программным путем, используя метод Hide объекта UserForm.*

### **3.4. Создание средств запуска приложения VBA из Windows**

а) в окне Project - VBAProject() / MicrosoftExcel Objects / Эта книга  
записать процедуру:

```
Private Sub Workbook_WindowActivate(ByVal Wn As Excel.Window)  
' Открытие нужной UserForm, например, UserForm4  
UserForm1.Show  
End Sub
```

б) для запуска приложения с рабочего стола Windows, создать на рабочем столе ярлык файла Excel, в котором создано приложение.

### **3.5. Упражнения 1**

1. Создайте на VBA функцию пользователя «Стоимость».
2. Создайте на VBA приложение «Калькулятор», используя приведенные выше вид UserForm (рис. 3.2) и текст процедуры.
3. Создайте на VBA приложение «Стоимость товара с учетом скидки», используя созданную в п.1 функцию «Стоимость».

*Указания по выполнению.*

При разработке приложения создайте UserForm по следующему образцу (рис. 3.3):

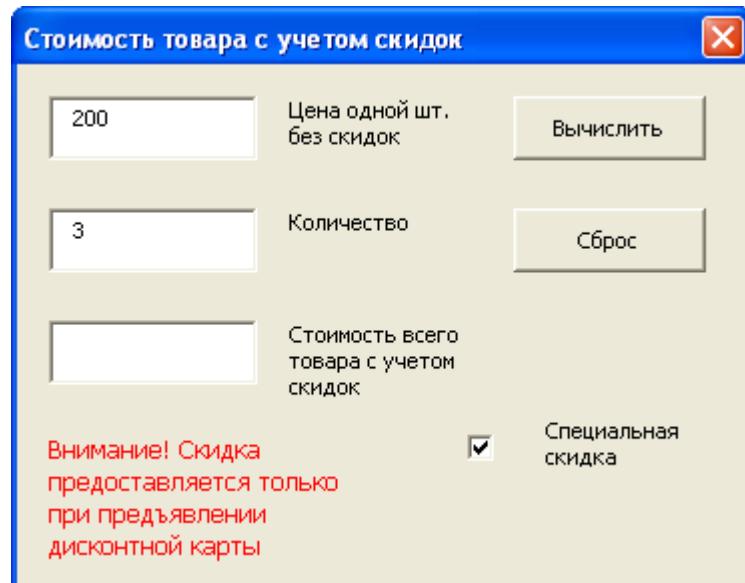


Рис. 3.3.

Для вычисления цены товара с учетом скидок используйте функцию «Стоимость». Надпись «Внимание! Скидка предоставляется только при предъявлении дисконтной карты» сделайте видимой только при установке флашка «Специальная скидка».

4. Создайте на VBA приложение "Расчет маргинальной процентной ставки".

*Условие задачи.* Клиент просит дать ссуду размером Р с условием возвращения в течение N лет, при размере одной выплаты А (конечно,  $A \cdot N > P$ ). Есть альтернатива - не давать ссуду клиенту, а положить деньги в банк под i процентов годовых.

*Требуется рассчитать:*

- какую сумму нужно положить в банк, чтобы получить (ежегодно!) такой же доход, как и в случае предоставления ссуды;
- под какой минимальный процент (это и есть маргинальная процентная ставка) можно положить в банк сумму денег, равную ссуде, чтобы получить тот же самый доход, что и при предоставлении ссуды.

#### Указания по выполнению.

- При выполнении п.а) используйте стандартную функцию PV(i,N,-A):  
 $p1=Application.PV(i,n,-a)$
- При составлении текста процедуры расчета используйте следующие типовые фрагменты:
  - Описание типов переменных Dim i As Double, Dim n As Integer;

б) функции чтения из окон ввода числовых значений с преобразованием текстового представления числа в число в формате Integer или Double:

```
i=CInt(TextBox1.Text)
```

```
p=CDbl(TextBox2.Text)
```

в) функции переформатирование вычисленных значений из числовой формы в текстовую и вывод их в диалоговые окна:

```
TextBox3.Text = CStr(Format(p,"Fixed"))
```

3. Для вычисления маргинальной процентной ставки создайте программным путем таблицу:

Табл. 3.1.

	A	B
1		
2	Число выплат	n
3	Размер ссуды	p
4	Размер одной выплаты	a
5	Процентная ставка	i
6	Текущий объем ссуды	=ПС(B5;B2;-B4)
7	Маргинальная процентная ставка	i
8	Маргинальный чистый объем ссуды	=ПС(B7;B2;-B4)

Собственно вычисление маргинальной процентной ставки произведите при использовании метода GoalSeek объекта Range.

При разработке приложения создайте UserForm по следующему образцу (рис. 3.4):

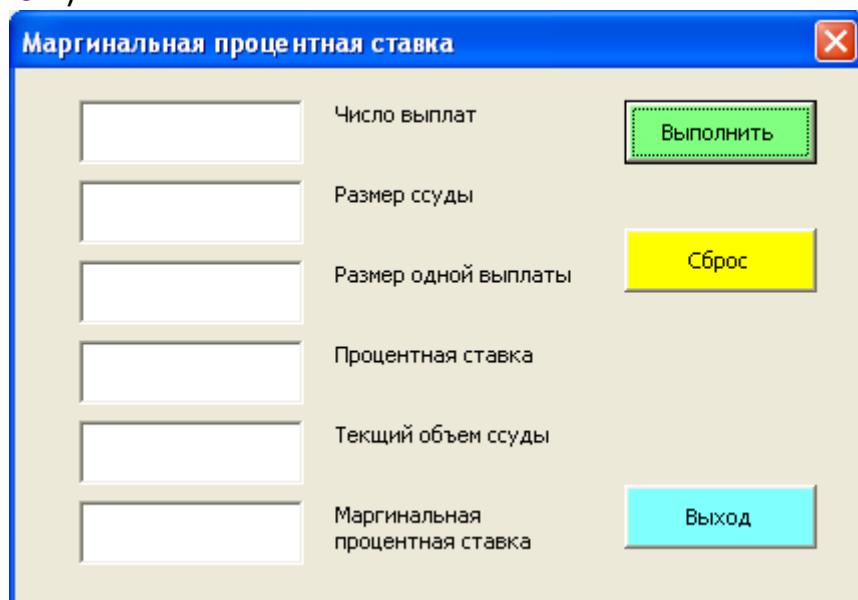


Рис. 3.4.

Создайте на VBA приложение для анализа доходов от издательской деятельности по разделу упражнений «Решение задач. Подбор параметра».

*Примечание. Используйте при создании приложения элемент управления ComboBox, МЕТОД AddItem при заполнении, СВОЙСТВО ListRow для задания количества элементов списка.*

Пример. ComboBox1.AddItem "КолЭкз"

ComboBox1.ListRow=4

#### Указания по выполнению

При разработке приложения создайте UserForm по следующему образцу (рис. 3.5):

Здесь «Расчет доходов/расходов», «Подбор параметра» и «Сброс» - кнопки управления (Button), «Фактор» - поле со списком (ComboBox), остальные элементы - окна ввода/вывода (TextBox).

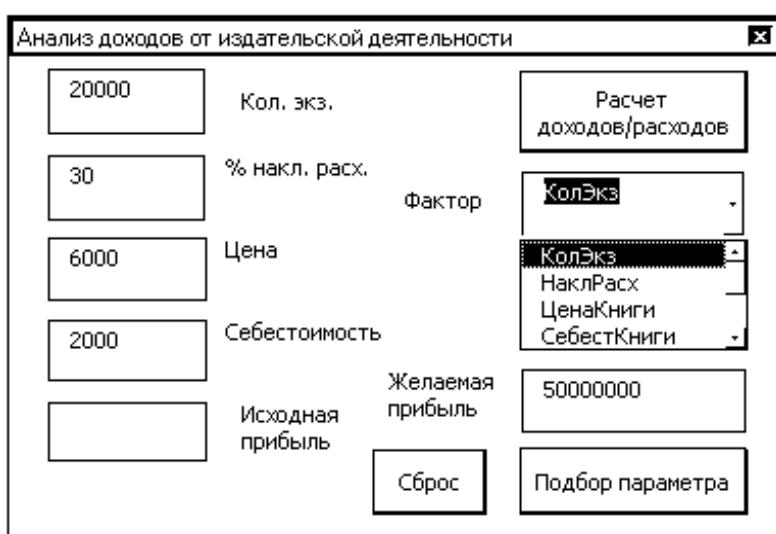


Рис. 3.5

Программа создаваемого приложения должна обеспечить:

а) создание таблицы :

Табл. 3.2.

	A	B
1	Расходы/доходы от издания книги	
2		
3		
4	Количество экземпляров	20 000
5	Доход	=B17*B4
6	Себестоимость	=B18*B4
7	Валовая прибыль	=B5-B6

8	% накладных расходов	30
9	Затраты на зарплату	=250*B4
10	Затраты на рекламу	=50*B4
11	Накладные расходы	=B5*B8/100
12	Валовые издержки	=B11+B9+B10
13		
14	Прибыль от продукции	=B7-B12
15		
16		
17	Цена продукции	6000
18	Себестоимость продукции	2000

6) подбор параметра в ячейке В14 (выяснить условия получения определенной заданной прибыли за счет всех возможных изменений факторов: тиража, уровня накладных расходов, цены и себестоимости).

Заполнение списка ComboBox производите программно, с помощью отдельной процедуры в модуле по следующему образцу:

```
Private Sub UserForm_Initialize()
With ComboBox1
    .AddItem "КолЭкз"
    .AddItem "НаклРасх"
    .AddItem "ЦенаКниги"
    .AddItem "СебестКниги"
    .ListRows = 4
End With
End Sub
```

При составлении программы рекомендуется воспользоваться примерами, приведенными в разделе «Основные средства VBA» и приведенными ниже:

а) Чтение текста из окна ввода и преобразование его в числовой формат (Integer):

```
k = CInt(TextBox2.Text)
```

б) Запрет ввода в окно, предназначенное для вывода:

```
TextBox6.Enabled = False
```

в) Назначение функции кнопки CommandButton1 клавише Enter:

```
With CommandButton1
```

```
    .Default = True
```

```
End With
```

г) Установка текста всплывающей подсказки для кнопки CommandButton1:

```
With CommandButton1
    .ControlTipText = "Поиск значения"
```

```
End With
```

Дополните разработанные приложения по п. 1-5, предусмотрев в них следующие дополнительные средства:

- запрет ввода текста в окна, предназначенные для вывода (свойство Enabled=False);
- всплывающие таблички с комментариями к элементам управления на панели UserForm (свойство ControlTipText);
- кнопки для закрытия панелей UserForm (используйте метод Hide);
- встроенные диалоговые окна InputBox и MsgBox;
- присвоение клавишам Enter и Cancel функций кнопок на панели UserForm (свойства Default и Cancel соответственно);
- дополнительную процедуру, выполняющую предыдущую функцию при инициализации UserForm (событие Initialize).

### ***3.6. Обработка и перехват ошибок в программах на VBA***

При работе с приложениями, предусматривающими ввод исходных данных, типичными являются следующие ошибки пользователя:

- ◆ отсутствие исходных данных (запуск приложения при вводе не всех исходных данных);
- ◆ неверный формат введенных пользователем исходных данных;
- ◆ ввод значений исходных данных вне допустимого диапазона.

Обработка ошибок заключается в формировании диагностических сообщений пользователю и аварийный выход из процедуры. При разработке приложения на VBA необходимо предусматривать программные средства для обработки таких ошибок.

Средства VBA анализируют ошибки и прерывают работу программы с выдачей сообщения о номере ошибки, хотя и не во всех случаях. Полезно запрограммировать перехват ошибок, обнаруженных средствами VBA, и

обработку этих ошибок с тем, чтобы предотвратить аварийный останов программы и переход в режим отладки.

### Отсутствие исходных данных. Неверный формат.

Если в окно ввода ошибочно не введено число, которое используется в программе, или введено число, но не в том формате, а программа запущена, то при выполнении в программе инструкции типа `a=CDbl(TextBox1.Text)` произойдет ошибка "несоответствие типов".

Необходимо включить в **начало** программы инструкции проверки типов, формирования сообщения о причине ошибки и выхода из процедуры, чтобы проверка типов происходила до момента преобразования текстовой строки в число, дату, массив и т. д.

В VBA имеются следующие функции проверки типов: `IsArray`, `IsDate`, `IsEmpty`, `IsError`, `IsNull`, `IsNumeric`, `IsObject`.

Синтаксис: `IsXXXX(переменная)`

Пример программы проверки содержимого окна ввода на соответствие числовому формату Double:

```
If IsNumeric(TextBox1.Text)= False Then
    MsgBox("Ошибка формата при вводе")
    TextBox1.SetFocus
    Exit Sub
End If
```

В результате при ошибке формата при вводе произойдет выход из процедуры (но изображение UserForm сохранится) и курсор будет установлен на окно TextBox1, в котором была сделана ошибка.

### Ввод значений вне допустимого диапазона.

Необходимо предупреждать пользователя разработанного приложения о выходе вводимого им значения из допустимого диапазона в тех случаях, когда это приводит к аварийной остановке программы или к результатам, не имеющим смысла.

Эту проверку необходимо производить после проверки соответствия типов. При проверке производится чтение содержимого окна ввода, преобразование

текста в нужный формат и собственно проверка соответствия полученного значения диапазону.

Пример проверки того, что введенное числовое значение - не ноль:

```
If CDbl(TextBox1.Text) = 0 Then
    MsgBox "Значение не должно быть нулевым!"
    TextBox1.SetFocus
Exit Sub
End If
```

В результате при ошибке диапазона при вводе произойдет выход из процедуры (но изображение UserForm сохранится) и курсор будет установлен на окно TextBox1, в котором была сделана ошибка.

Примеры преобразования в другие форматы: CBool, CByte, CCur, CDate, CDec, CInt, CLng, CSng, CVar, CStr - Boolean, Byte, Currency, Date, Decimal, Integer, Long, Single, Variant, String.

Ниже приведен пример рассмотренной ранее процедуры (полное количество процедур равно количеству кнопок управления) с обработкой ошибок из приложения «Калькулятор»:

```
Private Sub CommandButton3_Click()
If IsNumeric(TextBox2.Text) = False Then
    MsgBox ("Ошибка формата при вводе")
    TextBox2.SetFocus
Exit Sub
End If
x = CDbl(TextBox2.Text)
If x > 1 Then
    MsgBox "Значение не должно быть больше 1!", vbExclamation
    TextBox2.SetFocus
Exit Sub
End If
Range("B3").Value = x
Range("B4").Formula = "=ACOS(B3)"
y = Range("B4").Value
TextBox1.Text = CStr(Format(y, "#####0.0#####"))
End Sub
```

### **3.7. Упражнения 2**

1. Дополните программу «Расчет маргинальной процентной ставки» средствами обработки ошибок:

а) выход из процедуры с установкой фокуса на окно ввода (TextBox1) при ошибке в формате данных;

б) выход из процедуры с установкой фокуса на окно ввода при ошибке в данных (если сумма выплат оказывается меньше ссуды).

2. Дополните программу «Расчет доходов от издательской деятельности» средствами обработки ошибок ввода данных в окна ввода.

### **Перехват ошибок**

В случае возникновения ошибок происходит аварийный останов программы, переход в режим отладки и формирование сообщения о типе ошибки.

Каждая ошибка в VBA имеет свой код, например:

6 – переполнение;

9 – индекс выходит за пределы допустимого диапазона;

11 – деление на нуль;

13 – несоответствие типа.

программы с сообщением о типе ошибки.

Желательно предусматривать в программе обнаружение таких ошибок до того, как их обнаружит VBA, избежать аварийного останова программы и перехода в режим отладки, а также создать более понятное пользователю сообщение о типе ошибки и возможных действиях по ее устранению.

Особенностью многих ошибок является то, что нельзя указать - где конкретно произошла ошибка. Например, при вычислении сложного выражения причина переполнения может быть в любом из элементов выражения. В связи с этим в такой ситуации для того, чтобы избежать аварийного останова программы, необходимо при обнаружении ошибки заменять все параметры выполняемой процедуры на такие, при которых ошибка гарантированно не может возникнуть. Например, при программировании операции деления ошибка переполнения может возникнуть в одном из трех случаев:

- недопустимо большое значение имеет числитель;
- недопустимо малое значение имеет знаменатель;

- допустимое, но большое значение имеет числитель и одновременно допустимое, но малое значение имеет знаменатель.

В этом случае при возникновении ошибки переполнения можно в программе обработки ошибок предусмотреть замену числителя и знаменателя на такие значения, при которых ошибка переполнения не может возникнуть, например, сделать числитель и знаменатель равными единице.

Ниже приведен пример такой программы, решающей задачу деления чисел. В программе предусмотрены как обработка, так и перехват ошибок.

### Пример

```

Private Sub CommandButton1_Click()
Dim X, Y, Z As Single
'Передача управления на обработчик ошибок
On Error GoTo Обработка
If IsNumeric(TextBox1.Text) = False Then
    MsgBox "Ошибка в числителе", vbInformation, "Повторите ввод"
    TextBox1.SetFocus
    Exit Sub
End If
If IsNumeric(TextBox2.Text) = False Then
    MsgBox "Ошибка в знаменателе", vbInformation, "Повторите ввод"
    TextBox2.SetFocus
    Exit Sub
End If
Y = CDbl(TextBox1.Text)
X = CDbl(TextBox2.Text)
Z = Y / X
TextBox3.Text = CStr(Z)
Exit Sub

'Обработчик ошибок
' 6-переполнение, 11-деление на нуль, 13-несоответствие типов

Обработка:
Select Case Err.Number

```

```
Case Is = 6
MsgBox "Произошла ошибка переполнения", vbInformation
TextBox1.Text = 1
TextBox2.Text = 1
Y = 1
X = 1
Resume
```

```
Case Is = 11
MsgBox "Произошла ошибка деления на нуль", vbInformation
TextBox1.Text = 1
TextBox2.Text = 1
Y = 1
X = 1
Resume
```

```
Case Is = 13
MsgBox "Произошла ошибка несоответствия типов", vbInformation
TextBox1.Text = 1
TextBox2.Text = 1
Y = 1
X = 1
Resume
```

```
Case Else
MsgBox "Произошла ошибка: " & Err.Description & vbInformation
TextBox1.Text = 1
TextBox2.Text = 1
Y = 1
X = 1
Resume
End Select
End Sub
```

### **3.8. Использование макросов**

При написании программы на VBA, включающей программирование сложных действий с таблицами Excel 2016, можно использовать макросы. Для этого нужно выполнять такие действия в Excel 2016, включив предварительно макрорекордер. Полученные в результате макросы, представляющие по существу программы на языке Visual Basic, могут быть вставлены в программу. Ниже приведен пример использования макросов.

#### **Приложение "Учет продаж"**

В данном приложении UserForm используется для ведения базы данных и составления итоговых отчетов. При создании программы на VBA используются макросы, созданные в Excel 2016.

Поставленная задача - учет продаж и финансовых поступлений от предприятий, занимающихся реализацией произведенной продукции, - решается путем создания на языке VBA базы данных в Excel 2016.

Создание отчетов по фирмам-потребителям и по датам может быть также выполнено по программе, реализованной на VBA, но более просто выполнить все необходимые для этого действия в Excel 2016, включив MacroRecorder. Текст макроса, который будет при этом получен, можно будет включить в программу на VBA.

Табл. 3.3.

Дата	Наим.орг	Кол1	Цена	Сумма	Кол2	Цена	Сумма	Всего	Дата	Оплата	Остаток
01.янв	ВИЛАРД	50	2,45	122,5			0	122,5	15.янв	100	22,5
01.янв	КАРАТ	80	2,4	192			0	192		0	192
01.янв	КОНДАС	50	2,45	122,5	20	4,4	88	210,5	05.янв	200	10,5
01.янв	ЛИГА	20	2,5	50			0	50			50
02.янв	ЛОРА	100	2,6	260	10	4,3	43	303	07.янв	303	0
02.янв	СОЛО	50	2,3	115			0	115		0	115
02.янв	ТИГ	50	2,5	125			0	125		0	125
03.янв	АЛЬЯНС	50	2,5	125			0	125		0	125
03.янв	АННА-МАРИЯ	200	2,4				0	480		0	480
03.янв	АЯКС	50	2,3	115			0	115		0	115
03.янв	ВИЛАРД	100	2,3	230			0	230		0	230
03.янв	КОНДАС	100	2,5	250			0	250		0	250
03.янв	СТО	50	2,5	125			0	125		0	125
03.янв	ТИГ	100	2,45	245			0	245	10.янв	200	45

**Указания по выполнению**

- UserForm рекомендуется создать по приведенной ниже форме рис. 3.6 и с приведенным набором окон ввода/вывода и кнопок управления.
- При программном создании в Excel 2016 приведенной выше таблицы используйте следующие объекты, методы, свойства и события VBA:
  - Заполнение строки заголовков таблицы базы данных:  
`Range("A1:L1").Value = Array("Дата", "НаимОрг.", "Кол1", "Цена1", "Сумма1", "Кол2", "Цена2", "Сумма2", "Всего", "Дата_опл.", "Оплата", "Остаток")`
  - Чтение из окон графического интерфейса, присваивание или вычисление значений переменных «Дата», «Наим.орг.», «Кол1» и др. в программе
  - Определение текущей пустой строки БД:  
`НомерСтроки = Application.CountA(ActiveSheet.Columns(1)) + 1`
  - Внесение очередной записи в БД:  
`With ActiveSheet  
 .Cells(НомерСтроки, 1).Value = Дата  
 .....  
End With`

*Примечания*

- в именах не должны содержаться символы точки и запятой;
- значения Цена1 и Цена2 возьмите равными 30,0 и 40,0 соответственно;
- значения Сумма1, Сумма2, Всего должны быть вычислены в программе.

3. Для создания модулей формирования отчетов после создания базы данных создайте макросы в Excel 2016, используя MacroRecorder, затем скопируйте тексты макросов через буфер обмена.

Рис. 3.6.

**3.9. Защита программного кода приложений пользователя**

- открыть окно программного кода;
- [Tools] – [VBA Project Properties...] – [Protections];
- ввести пароль и подтверждение пароля в окна «Password» и «Confirm Password» всплывающего диалогового окна «VBA Project – Project Properties»

### **3.10. Примеры выполнения задачий**

#### **Примеры выполнения задачий 1 – 6.**

##### **Приложение «Расчет стоимости товара при оптовой продаже с учетом скидок»**

```
Private Sub CheckBox1_Click()
If CheckBox1.Value = True Then
    Label5.Visible = True
Else
    Label5.Visible = False
End If
End Sub
```

```
Private Sub CommandButton1_Click()
    Цена = CDbl(TextBox1.Text)
    Количество = CDbl(TextBox2.Text)
    If CheckBox1.Value = True Then
        Скидка = 1
        Label5.Visible = True
    Else
        Скидка = 0
        Label5.Visible = False
    End If
    СтоимостьТовара = Стоимость(Цена, Количество, Скидка)
    TextBox3.Text = CStr(Format(Стоимость Товара, "Fixed"))
End Sub
```

```
Private Sub CommandButton2_Click()
    TextBox1.Text = ""
    TextBox2.Text = ""
    TextBox3.Text = ""
End Sub
```

```
Private Sub UserForm_Initialize()
    Label5.Visible = False
End Sub
```

## Приложение "Расчет маргинальной процентной ставки"

### Алгоритм расчета.

1. Создаем программным путем таблицу в рабочей книге Excel 2016:  
**B2** – число выплат - (n);  
**B3** – размер ссуды, которую запрашивает клиент - (p) ;  
**B4** – размер ежегодной выплаты, которую обещает клиент - (a);  
**B5** – процентная ставка банка, под которую можно было бы положить деньги, вместо того, чтобы отдавать клиенту – (i);  
**B7** – клетка для последующего расчета маргинальной процентной ставки, вначале в нее заносится начальное приближение – (i);  
**B6, B8** – клетки, в которые заносится формула PV со ссылкой на клетки B5 и B7 – i, B2 – n, B4 – a, для вычисления размера ссуды, положив которую в банк под фиксированный процент i, можно в течение n лет ежегодно получать выплаты размером a. Excel 2016 произведет вычисление этого значения сразу же после внесения формулы.
2. Используя в качестве целевой ячейки B8, произведем подбор параметра (функция GoalSeek, условие B8=p, изменяемая ячейка B7, в которой к этому моменту находится начальное приближение - i). В результате в ячейке B7 будет получено значение маргинальной процентной ставки.

Таким образом, в ячейке B6 в результате будет находиться текущий объем ссуды (сколько нужно положить в банк под фиксированный процент i, чтобы получить столько же, сколько предлагает клиент), в ячейке B8 – маргинальный чистый объем ссуды (он равен размеру ссуды, которую просит клиент), в ячейке B7 – маргинальная процентная ставка.

```
Private Sub CommandButton1_Click()
Dim n As Integer
Dim p As Double
Dim A As Double
Dim i As Double
Dim iMarg As Double
```

```

Dim pPure As Double
n = CInt(TextBox1.Text)
p = CDbl(TextBox2.Text)
A = CInt(TextBox3.Text)
i = CInt(TextBox4.Text) / 100
'Создание таблицы на листе Excel
With ActiveSheet
    .Range("A2").Value = "Число выплат"
    .Range("A3").Value = "Размер ссуды"
    .Range("A4").Value = "Размер одной выплаты"
    .Range("A5").Value = "Процентная ставка"
    .Range("A6").Value = "Текущий объем ссуды"
    .Range("A7").Value = "Маргин. процентная ставка"
    .Range("A8").Value = "Маргин. чистый текущий объем ссуды"
    .Range("B2").Value = n
    .Range("B3").NumberFormat = "#,##0$"
    .Range("B3").Value = p
    .Range("B4").NumberFormat = "#,##0$"
    .Range("B4").Value = A
    .Range("B5").NumberFormat = "0,00%"
    .Range("B5").Value = i
    'Ввод начального приближения для марг. проц. ставки
    .Range("B7").NumberFormat = "0,00%"
    .Range("B7").Value = i
    .Range("B6").Formula = "=PV(B5,B2,-B4)"
    .Range("B8").Formula = "=PV(B7,B2,-B4)"
    .Range("B8").GoalSeek Goal:=p, ChangingCell:=.Range("B7")
    iMarg = .Range("B7").Value
End With
'Автоподгонка ширины столбцов А и В
ActiveSheet.Columns("A:B").Select
Selection.Columns.AutoFit
'Вычисление текущего объема ссуды и марг. проц. ставки
pPure = Application.PV(i, n, -A)
TextBox5.Text = CStr(Format(pPure, "Fixed"))
TextBox6.Text = CStr(Format(iMarg * 100, "Fixed"))
End Sub

```

```
Private Sub CommandButton2_Click()
    TextBox1.Text = ""
    TextBox2.Text = ""
    TextBox3.Text = ""
    TextBox4.Text = ""
    TextBox5.Text = ""
    TextBox6.Text = ""
End Sub
```

```
Private Sub CommandButton3_Click()
    UserForm4.Hide
End Sub
```

```
Private Sub UserForm_Initialize()
    'Запрет ввода в окна 5 и 6, предназначенные для вывода
    TextBox5.Enabled = False
    TextBox6.Enabled = False
    'Присвоение клавише <Enter> функции кнопки "Выполнить"
    With CommandButton1
        .Default = True
        .ControlTipText = "Расчет текущего объема ссуды и маргинальной
процентной ставки"
    End With
    With CommandButton2
        .Cancel = True
        .ControlTipText = "Очистка окон"
    End With
End Sub
```

### **Анализ доходов от издательской деятельности**

```
Private Sub UserForm_Initialize()
    With ComboBox1
        .AddItem "КолЭкз"
        .AddItem "НаклРасх"
        .AddItem "ЦенаКниги"
        .AddItem "СебестКниги"
    End With
End Sub
```

```
.ListRows = 4
End With
End Sub
```

'Обработка нажатия кнопки "Расчет доходов/расходов"

```
Private Sub CommandButton1_Click()
Dim pp As Double 'Прибыль от продукции
Dim k As Integer ' Количество экземпляров
Dim pnr As Integer ' % накл. расходов
Dim zp As Integer ' Цена книги
Dim sp As Integer ' Себестоимость книги

k = CInt(TextBox2.Text)
pnr = CInt(TextBox3.Text)
zp = CInt(TextBox4.Text)
sp = CInt(TextBox5.Text)
'Запрет ввода в окно "Исходная прибыль" предназначенное для вывода
TextBox6.Enabled = False
'Назначение функции кнопки CommandButton1 клавише <Enter>
'Установка текста всплывающей подсказки для кнопки CommandButton1
With CommandButton1
    .Default = True
    .ControlTipText = "Поиск значения фактора"
End With
With ActiveSheet
    .Range("A1:C20").Clear
    .Range("A4").Value = "Количество экземпляров"
    .Range("A5").Value = "Доход"
    .Range("A6").Value = "Себестоимость всего тиража "
    .Range("A7").Value = "Валовая прибыль"
    .Range("A8").Value = "% накладных расходов"
    .Range("A9").Value = "Затраты на зарплату"
    .Range("A10").Value = "Затраты на рекламу"
    .Range("A11").Value = "Накладные расходы"
    .Range("A12").Value = "Валовые издержки"
    .Range("A14").Value = "Прибыль от продукции"
```

```

.Range("A17").Value = "Цена одной книги"
.Range("A18").Value = "Себестоимость одной книги"
End With
ActiveSheet.Columns("A:A").Select
With Selection
    .ColumnWidth = 30
End With
With ActiveSheet
    .Range("B4").Value = k
    .Range("B8").NumberFormat = "0.00%"
    .Range("B8").Value = pnr
    .Range("B17").NumberFormat = "#,##0$"
    .Range("B17").Value = zp
    .Range("B18").NumberFormat = "#,##0$"
    .Range("B18").Value = sp
    .Range("B5").Formula = "=B17*B4"
    .Range("B6").Formula = "=B18*B4"
    .Range("B7").Formula = "=B5-B6"
    .Range("B9").Formula = "=250*B4"
    .Range("B10").Formula = "=50*B4"
    .Range("B11").Formula = "=B5*B8/100"
    .Range("B12").Formula = "=B9+B10+B11"
    .Range("B14").Formula = "=B7-B12"
    npp = .Range("B14").Value
    TextBox6.Text = CStr(Format(npp, "Fixed"))
End With
End Sub
'Обработка нажатия кнопки "Подбор параметра"
Private Sub CommandButton2_Click()
With ActiveSheet
    pp = CDbl(TextBox1.Text)
    If ComboBox1.Text = "КолЭкз" Then
        .Range("B14").GoalSeek Goal:=pp, ChangingCell:=.Range("B4")
        kk = .Range("B4").Value
        TextBox2.Text = CStr(Format(kk, "Fixed"))
    End If
    If ComboBox1.Text = "НаклРасх" Then

```

```

.Range("B14").GoalSeek Goal:=pp, ChangingCell:=.Range("B8")
kk = .Range("B8").Value
TextBox3.Text = CStr(Format(kk, "Fixed"))
End If
If ComboBox1.Text = "ЦенаКниги" Then
    .Range("B14").GoalSeek Goal:=pp, ChangingCell:=.Range("B17")
    kk = .Range("B17").Value
    TextBox4.Text = CStr(Format(kk, "Fixed"))
End If
If ComboBox1.Text = "СебестКниги" Then
    .Range("B14").GoalSeek Goal:=pp, ChangingCell:=.Range("B18")
    kk = .Range("B18").Value
    TextBox5.Text = CStr(Format(kk, "Fixed"))
End If
End With
End Sub
' Обработка нажатия кнопки "Сброс"
Private Sub CommandButton3_Click()
    TextBox2.Text = ""
    TextBox3.Text = ""
    TextBox4.Text = ""
    TextBox5.Text = ""
    TextBox6.Text = ""
    TextBox1.Text = ""
End Sub

```

**Примеры выполнения заданий раздела «Обработка ошибок в программах на VBA»**

**Приложение "Расчет маргинальной процентной ставки"**

```

Private Sub CommandButton1_Click()
Dim n As Integer
Dim p As Double
Dim A As Double
Dim i As Double
Dim iMarg As Double

```

```
Dim pPure As Double
```

```
' Выход из процедуры с установкой фокуса в окно TextBox1 при ошибке в
'формате данных
```

```
If IsNumeric(TextBox1.Text) = False Then
```

```
    MsgBox "Ошибка в формате данных", vbCritical, "Повторите ввод"
```

```
    TextBox1.SetFocus
```

```
    Exit Sub
```

```
End If
```

```
' Выход из процедуры с установкой фокуса в окно TextBox2 при ошибке в
'формате данных
```

```
If IsNumeric(TextBox2.Text) = False Then
```

```
    MsgBox "Ошибка в формате данных", vbCritical, "Повторите ввод"
```

```
    TextBox2.SetFocus
```

```
    Exit Sub
```

```
End If
```

```
' Выход из процедуры с установкой фокуса_
```

```
' в окно TextBox3 при ошибке в формате данных
```

```
If IsNumeric(TextBox3.Text) = False Then
```

```
    MsgBox "Ошибка в формате данных", vbCritical, "Повторите ввод"
```

```
    TextBox3.SetFocus
```

```
    Exit Sub
```

```
End If
```

```
' Выход из процедуры с установкой фокуса в окно TextBox1 при ошибке
```

```
' в формате данных
```

```
If IsNumeric(TextBox4.Text) = False Then
```

```
    MsgBox "Ошибка в формате данных", vbCritical, "Повторите ввод"
```

```
    TextBox4.SetFocus
```

```
    Exit Sub
```

```
End If
```

```
n = CInt(TextBox1.Text)
```

```
p = CDbl(TextBox2.Text)
```

```
A = CInt(TextBox3.Text)
```

```
i = CInt(TextBox4.Text) / 100
```

```
'Выход из процедуры с установкой фокуса в окно TextBox1
```

```
'если сумма выплат оказывается меньше ссуды
```

```

If n * A < p Then
    MsgBox "Клиент собирается возвратить на " & CStr(Format(p - n * A, "Fixed"))
    & " меньше размера ссуды!", vbExclamation
    TextBox1.SetFocus
    Exit Sub
End If
'Создание таблицы на листе Excel
With ActiveSheet
    .Range("A2").Value = "Число выплат"
    .Range("A3").Value = "Размер ссуды"
    .Range("A4").Value = "Размер одной выплаты"
    .Range("A5").Value = "Процентная ставка"
    .Range("A6").Value = "Текущий объем ссуды"
    .Range("A7").Value = "Маргин. процентная ставка"
    .Range("A8").Value = "Маргин. чистый текущий объем ссуды"
    .Range("B2").Value = n
    .Range("B3").NumberFormat = "#,##0$"
    .Range("B3").Value = p
    .Range("B4").NumberFormat = "#,##0$"
    .Range("B4").Value = A
    .Range("B5").NumberFormat = "0,00%"
    .Range("B5").Value = i
    'Ввод начального приближения для марг. проц. ставки
    .Range("B7").NumberFormat = "0,00%"
    .Range("B7").Value = i
    .Range("B6").Formula = "=PV(B5,B2,-B4)"
    .Range("B8").Formula = "=PV(B7,B2,-B4)"
    .Range("B8").GoalSeek Goal:=p, ChangingCell:=.Range("B7")
    iMarg = .Range("B7").Value
End With
'Автоподгонка ширины столбцов А и В
ActiveSheet.Columns("A:B").Select
Selection.Columns.AutoFit
'Вычисление текущего объема ссуды и марг. проц. ставки
pPure = Application.PV(i, n, -A)
TextBox5.Text = CStr(Format(pPure, "Fixed"))
TextBox6.Text = CStr(Format(iMarg * 100, "Fixed"))

```

End Sub

```
Private Sub CommandButton2_Click()
    TextBox1.Text = ""
    TextBox2.Text = ""
    TextBox3.Text = ""
    TextBox4.Text = ""
    TextBox5.Text = ""
    TextBox6.Text = ""
End Sub
```

```
Private Sub CommandButton3_Click()
    UserForm4.Hide
End Sub
```

```
Private Sub UserForm_Initialize()
    'Запрет ввода в окна 5 и 6, предназначенные для вывода
    TextBox5.Enabled = False
    TextBox6.Enabled = False
    'Присвоение клавише <Enter> функции кнопки "Выполнить"
    With CommandButton1
        .Default = True
        .ControlTipText = "Расчет текущего объема ссуды и маргинальной
            процентной ставки"
    End With
    With CommandButton2
        .Cancel = True
        .ControlTipText = "Очистка окон"
    End With
End Sub
```

### ***Примеры выполнения заданий по разделу «Использование макросов»***

#### **Приложение "Учет продаж"**

```
Private Sub CommandButton1_Click()
    Dim Дата As Date
```

```
Dim НаимОрганиз As String * 20
Dim Кол1 As Integer
Dim Цена1 As Double
Dim Сумма1 As Double
Dim Кол2 As Integer
Dim Цена2 As Double
Dim Сумма2 As Double
Dim Всего As Double
Dim ДатаОпл As Date
Dim Оплата As Double
Dim Остаток As Double
'Определение текущей пустой строки БД
НомерСтроки = Application.CountA(ActiveSheet.Columns(1)) + 1
With UserForm6
    Дата = .TextBox1.Text
    НаимОрганиз = .ComboBox1.Text
    Кол1 = CInt(TextBox2.Text)
    Цена1 = 2.45
    Кол1 = CInt(TextBox2.Text)
    Кол2 = CInt(TextBox3.Text)
    Цена2 = 4.5
    Оплата = CDbl(TextBox6.Text)
    ДатаОпл = .TextBox7.Text
End With
'Внесение очередной записи в БД
With ActiveSheet
    .Cells(НомерСтроки, 1).Value = Дата
    .Cells(НомерСтроки, 2).Value = НаимОрганиз
    .Cells(НомерСтроки, 3).Value = Кол1
    .Cells(НомерСтроки, 4).Value = Цена1
    .Cells(НомерСтроки, 5).Value = Кол1 * Цена1
    .Cells(НомерСтроки, 6).Value = Кол2
    .Cells(НомерСтроки, 7).Value = Цена2
    .Cells(НомерСтроки, 8).Value = Кол2 * Цена2
    Всего = (Кол1 * Цена1) + (Кол2 * Цена2)
    .Cells(НомерСтроки, 9).Value = Всего
    .Cells(НомерСтроки, 10).Value = ДатаОпл
```

```
.Cells(НомерСтроки, 11).Value = Оплата
.Cells(НомерСтроки, 12).Value = Всего - Оплата
End With
TextBox4.Text = CStr(Всего)
TextBox5.Text = CStr(Всего - Оплата)
End Sub
Private Sub CommandButton2_Click()
<текст макроса 1>
End Sub
Private Sub CommandButton3_Click()
<текст макроса 2>
End Sub
Private Sub UserForm_Initialize()
'Заполнение поля со списком
With ComboBox1
.AddItem "ВИЛАРД"
.AddItem "КАРАТ"
.AddItem "КОНДАС"
.AddItem "ЛИГА"
.AddItem "ЛОРА"
.AddItem "СОЛО"
.AddItem "ТИГ"
.AddItem "АЛЬЯНС"
.AddItem "АННА-МАРИЯ"
.AddItem "АЯКС"
.AddItem "СТО"
.AddItem "ВОЛАНД"
.ListRows = 12
End With
'Если заголовки полей существуют, то выход из процедуры
If Range("A1").Value = "Дата" Then
Range("A2").Select
Exit Sub
End If
'Если заголовков полей нет - они будут создаваться
ActiveSheet.Cells.Clear
```

```
Range("A1:L1").Value = Array("Дата", "НайлОрганиз", "Кол1", "Цена1",
"Сумма1", "Кол2", "Цена2", "Сумма2", "Всего", "Дата_опл.", "Оплата",
"Остаток")
Range("B:B").ColumnWidth = 14
End Sub
Private Sub CommandButton4_Click()
UserForm6.Hide
End Sub
Private Sub CommandButton5_Click()
ActiveSheet.Cells.Clear
End Sub
```

*Примечание. Номера UserForm в примерах программ даны для определенности.*

## Приложение 1. Примеры методов и свойств объектов VBA в Excel.

### Объект Worksheet и семейство Worksheets

Метод	Действие	Пример
Activate	Активизация указанного рабочего листа	Worksheets(1).Activate
Add	Создание нового рабочего листа	ActiveWorkbook.Worksheets.Add
Delete	Удаление рабочего листа	Worksheets(1).Delete
Protect	Установка защиты активного листа от внесения в него изменений	ActiveSheet.Protect Password:="Мария"
Unprotect	Снятие защиты с активного листа	ActiveSheet.Unprotect Password:="Мария"
Copy	Копирование активного листа в другое место рабочей книги	Worksheets("Лист1").Copy_ After:=Worksheets("Лист3")

### Объект Range

Свойство	Действие	Пример
Value	Возвращает значение из ячейки Заносит в ячейку значение Заносит в ячейку текст	x=Range("B1").Value Range("B1").Value=y Range("B1").Value="Объем ссуды"
NumberFormat	Заносит в ячейку параметры формата	Range("B3").NumberFormat="0.00%"
Formula	Заносит в ячейку формулу	Range("B5").Formula="=A5*B17"
WrapText	Устанавливает режим ввода текста в ячейку с переносом по словам	With Range("B2") .Value="Компьютер Apple" .WrapText = True End With

### Объект Range

Метод	Действие	Пример
Select	Выделяет диапазон ячеек	Range("A1:G17").Select
AutoFit	Автоматически настраивает ширину помеченных строк /столбцов	Selection.Rows.AutoFit Selection.Columns.AutoFit
Clear	Очищает диапазон ячеек	Range("A1:G17").Clear
Copy	Копирует диапазон ячеек в другой диапазон	Range("A1:A4").Copy_ Destination:=Range("D1:D4") <sup>12</sup>
GoalSeek	Производит подбор параметра	Range("B8").GoalSeek Goal:=p_ ChangingCell:=Range("B7") <sup>13</sup>

<sup>12</sup> Знак «подчеркивание» требуется, если часть операторного выражения переносится в следующую строку.

<sup>13</sup> R8 – целевая ячейка, p – значение, которое в ней требуется получить, B7 – изменяемая ячейка

**Приложение 2. Примеры методов, свойств и событий элементов управления VBA в Excel**

### Объект UserForm

Метод	Действие	Пример
Show	Отображает UserForm на экране	UserForm1.Show
Hide	Скрывает UserForm	UserForm1.Hide

Событие	Действие	Пример заголовка процедуры
Initialize	Происходит при отображении формы на экране	Private Sub UserForm_ Initialize ()
Terminate	Происходит при закрытии формы	Private Sub UserForm_ Terminate ()

### Объект TextBox

Свойство	Действие	Пример
Text	Возвращает текст, содержащийся в поле	Цена=TextBox1.Text
Enabled	Разрешает/запрещает вносить изменения в поле (окно TextBox)	TextBox1.Enabled=True TextBox1.Enabled=False
WordWrap	Разрешает/запрещает перенос в поле по словам	TextBox1. WordWrap =True TextBox1. WordWrap =False
Visible	Разрешает/запрещает отображение TextBox на экране	TextBox1. Visible =True TextBox1. Visible =False

### Объект Label

Свойство	Действие	Пример
Caption	Возвращает текст, отображаемый в надписи	Label2.Caption="Функция"
WordWrap	Разрешает/запрещает перенос в надписи по словам	Label2. WordWrap =True Label2. WordWrap =False
Visible	Разрешает/запрещает отображение надписи на экране	Label2. Visible =True Label2. Visible =False

**Объект CommandButton**

<b>Свойство</b>	<b>Действие</b>	<b>Пример</b>
Caption	Возвращает текст, отображаемый на кнопке	CommandButton1.Caption="Выполнить"
Enabled	Разрешает/запрещает нажатие кнопки пользователем	CommandButton1.Enabled =True CommandButton1.Enabled =False
Visible	Разрешает/запрещает отображение кнопки на экране во время выполнения программы	CommandButton1.Visible =True CommandButton1.Visible =False
Default	Присваивает/отменяет присвоение клавише <Enter> функции кнопки	CommandButton1.Default =True CommandButton1.Default =False
Cancel	Присваивает/отменяет присвоение клавише <Esc> функции кнопки	CommandButton1.Cancel =True CommandButton1.Cancel =False
ControlTipText	Создает всплывающую табличку с надписью	CommandButton1. ControlTipText="После нажатия на кнопку ваш ответ не может быть изменен"

**Объект OptionButton**

<b>Свойство</b>	<b>Действие</b>	<b>Пример</b>
Caption	Создает надпись, отображаемую рядом с переключателем OptionButton	OptionButton1.Caption="Выполнить"
Enabled	Разрешает/запрещает пользование переключателем	OptionButton1.Enabled =True OptionButton1.Enabled =False
Visible	Разрешает/запрещает отображение переключателя на экране во время выполнения программы	OptionButton1.Visible =True OptionButton1.Visible =False

**Объект CheckBox**

<b>Свойство</b>	<b>Действие</b>	<b>Пример</b>
Caption	Создает надпись, отображаемую рядом с флагжком CheckBox	CheckBox1.Caption="Выполнить"
Enabled	Разрешает/запрещает пользование флагжком	CheckBox1.Enabled =True CheckBox1.Enabled =False
Visible	Разрешает/запрещает отображение флагжка на экране во время выполнения программы	CheckBox1.Visible =True CheckBox1.Visible =False

### **Приложение 3. Основные операторы языка VBA**

#### **Оператор присваивания**

Оператор присваивает значение выражения переменной, константе или свойству объекта. Для присвоения переменной ссылки на объект применяется инструкция *Set*.

Синтаксис	Пример
Переменная(постоянная, свойство объекта)=Выражение	X=15 Y=Sin(X) Set Область=Range("A1:D10")

#### **Оператор безусловного перехода GoTo**

Задает безусловный переход на указанную строку внутри процедуры. Обязательный аргумент Стока может быть меткой строки или номером строки

Синтаксис	Пример
GoTo Стока	GoTo ОбработчикОшибок

#### **Оператор условного перехода If Then Else**

Если условие принимает значение *True*, то выполняется оператор или группа операторов после *Then*, если *False*, то выполняется оператор или группа операторов после *Else*. Возможна конструкция без *Else*.

Синтаксис	Пример
If Условие Then <Операторы> Else <Операторы > End If	If Скидка =1 Then Стоимость = Цена*Количество Else Стоимость = Цена*Количество*0.95 End If

## Оператор варианта Select Case

Обеспечивает переход к оператору (группе операторов) с меткой, соответствующей значению аргумента выражение, следующего после *Case*.

Синтаксис	Пример
Select Case выражение Case значение 1 <группа операторов> Case значение 2 <группа операторов> ..... Case значение m <группа операторов> End Select	Select Case x Case 1 MsgBox "Ошибка в числителе" Case 2 MsgBox "Ошибка в знаменателе" End Select

## Оператор цикла For-Next

Повторяет выполнение оператора (группы операторов) пока переменная цикла (счетчик) изменяется от начального до конечного значения с указанным шагом. По умолчанию шаг равен единице. [] – необязательный элемент.

Синтаксис	Пример
For Счетчик=Начало To Конец [Step Шаг] <Операторы> [Exit For] Next Счетчик	Dim Y(100) As Double For i=1 To 100 Y(i)=Sin(i/20) Next i

## Оператор цикла While – Wend

Повторяет выполнение оператора (группы операторов) пока заданное условие имеет значение True

Синтаксис	Пример
While Условие <Операторы> Wend	Dim Y(100) As Double i=1 While i<100 Y(i)=Sin(i/20) i=i+1 Wend

## Оператор цикла Do Until – Loop

Повторяет выполнение оператора (группы операторов) пока заданное условие имеет значение *True*. Условие проверяется после выполнения операторов. Альтернативный способ выхода из цикла предоставляет оператор *Exit Do*.

Синтаксис	Пример
<pre>Do [Until Условие] &lt;Операторы&gt; [Exit Do] &lt;Операторы&gt; Loop</pre>	<pre>Dim Y(100) As Double i=1 Do Until i&gt;100 Y(i)=Sin(i/20) i=i+1 Loop</pre>

## Оператор цикла Do While - Loop

Повторяет выполнение оператора (группы операторов) пока заданное условие имеет значение *True*. Условие проверяется после выполнения операторов. Альтернативный способ выхода из цикла предоставляет оператор *Exit Do*.

Синтаксис	Пример
<pre>Do [While Условие] &lt;Операторы&gt; [Exit Do] &lt;Операторы&gt; Loop</pre>	<pre>Dim y(100) As Double i=1 Do While i&lt;100 y(i)=Sin(i/20) i=i+1 Loop</pre>

## Оператор цикла Do - Loop Until

Повторяет выполнение оператора (группы операторов) пока заданное условие имеет значение *True*. Условие проверяется после выполнения операторов. Альтернативный способ выхода из цикла предоставляет оператор *Exit Do*.

Синтаксис	Пример
<pre>Do &lt;Операторы&gt; [Exit Do] &lt;Операторы&gt; Loop [While Условие]</pre>	<pre>Dim y(100) As Double i=1 Do Y[i]=Sin(i/20) i=i+1 Loop Until i&lt;100</pre>

## Типы данных

Тип данных	Название	Размер (байт)	Диапазон значений
Byte	Байт	1	От 0 до 255
Boolean	Логический	2	True или False
Integer	Целое число	2	От -32768 до 32767
Long	Длинное целое число	4	От -2 147 483 648 до 2 147 483 647
Single	Число с плавающей запятой обычной точности	4	От -3,402823E38 до -1,401298E-45 и От 16401298E-45 до 3,402823E38
Double	Число с плавающей запятой двойной точности	8	От -1,79769313486232E308 До -4,94065645841247E-324 и От 4,94065645841247E-324 До 1,79769313486232E308
Currency	Денежный	8	От -922 337 203 685 477,5808 До 922 337 203 685 477,58
Decimal	Масштабируемое целое число	14	+/- 79228162514264337593543950335 С 28 знаками справа от запятой Минимальное ненулевое значение имеет вид +/- 0, 00000000000000000000000000000001
Date	Дата и время	8	От 1 января 100г. до 31 декабря 9999г.
String	Строка переменной длины	10+длина строки	От 1 до приблизительно 2 миллиардов
String	Строка постоянной длины	Длина строки	От 1 до приблизительно 65400
Variant	Числовые подтипы	16	Любое числовое значение вплоть до границ диапазона для типа Double
Variant	Строковые подтипы	22+длина строки	Как для строки (String) переменной длины
Тип данных определяется пользователем с помощью ключевого слова Type	Пользовательский	Объем определяется элементами	Диапазон каждого элемента определяется его типом данных

## 4.Access 2016

Приложение ACCESS позволяет создавать простые базы данных (БД) в виде таблицы и реляционные базы данных и работать с ними (производить сортировку, фильтрацию, формировать многокритериальные запросы к БД и получать ответы на запросы в форме отчетов).

Для создания реляционной БД с помощью ACCESS необходимо выполнить следующие действия:

1. Выполнить аналитическое проектирование информационно - логической модели данных, результатом которого должно стать представление БД в форме набора взаимосвязанных таблиц.
2. Создать и сохранить файл базы данных.
3. Сформировать структуру таблиц, составляющих БД (Ввести названия и типы всех полей).
4. Установить связи между таблицами.
5. Создать формы для удобного заполнения и редактирования таблиц БД.
6. Заполнить таблицы БД данными с помощью созданных форм.
7. Спроектировать необходимые запросы и отчеты.

### 4.1. Объекты Access

Access ориентирована на работу с объектами, к которым относятся: таблицы базы данных, запросы, а также объекты приложений: формы, отчеты, макросы, модули.

Основными объектами Access являются таблицы.

Запуск программы Microsoft Access в WINDOWS 10/8/7 можно произвести из главного меню:

[Пуск] – [Все приложения] – [Access 2016].

В результате откроется окно Access (рис. 4.1).

Доступ к объектам базы данных (таблицам, формам, отчетам, запросам, макросам и другим объектам) производится из Области переходов окна Access 2016 (рис. 4.2).

#### 4.Access 2016

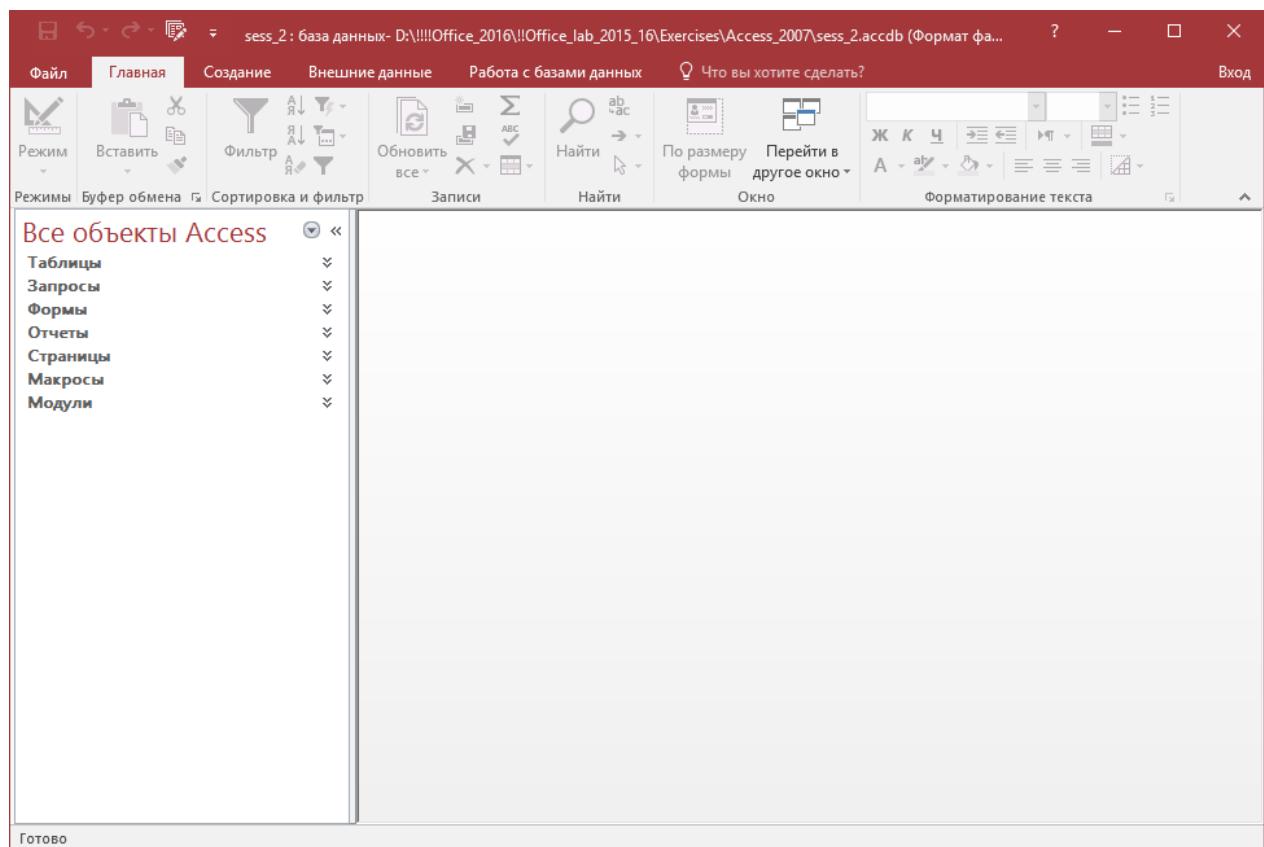


Рис. 4.1.

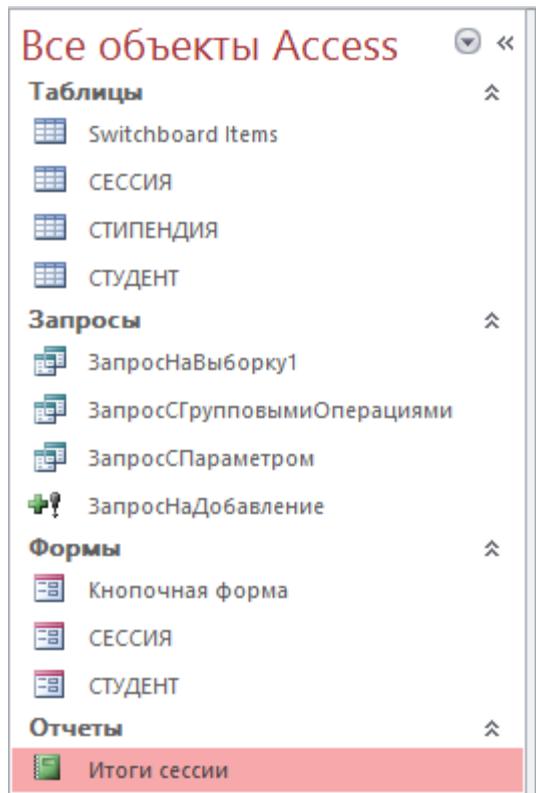


Рис. 5.

Основными объектами базы данных Microsoft Office Access 2016 являются:

- таблицы (Tables);
- формы (Forms);
- отчеты (Reports);
- запросы(Queries);
- Макросы (Macros);
- Модули (Modules);

### **Для чего нужны таблицы?**

Таблицы являются средствами хранения данных

### **Для чего нужны формы?**

- формы представляют собой более удобное и наглядное средство для ввода и редактирования данных в больших таблицах;

- b) формы позволяют вводить и просматривать данные сразу не из одной, а из нескольких связанных таблиц;
- c) формы, в отличие от таблиц, позволяют просматривать содержимое полей Длинный текст и OLE.

### **Для чего нужны отчеты?**

- a) отчет может быть составлен на основе данных из нескольких связанных таблиц;
- b) при создании отчета можно использовать разнообразные средства графического оформления.

### **Для чего нужны запросы?**

- a) запросы обеспечивают простой доступ к определенному подмножеству записей одной или нескольких таблиц;
- b) в запросе можно создать вычисляемые поля, которых нет в таблицах, указать условия сортировки и отбора записей;
- c) на основе запроса можно производить не только выборку, но удаление и обновление записей.

### **Для чего нужны макросы?**

Макросы предназначены (их удобно использовать) для автоматизации исполнения последовательности рутинных операций с БД, например:

- a) автоматического открытия и/или печати нескольких форм и отчетов;
- b) автоматического экспортования каких-либо данных из БД в другой формат, например, в Dbase;
- c) командой макроса RunCode <имя функции> можно запускать на исполнение функции, написанные на языке VBA;

Макросы могут запускаться "по кнопке" или при возникновении события.

## **Для чего нужны модули?**

Модули (функции, написанные на языке VBA) позволяют существенно расширить количество программируемых действий с базой данных, в том числе:

- a) получать и вносить в базу данных данные из файлов;
- b) запускать на исполнение .exe модули (команда Shell “<полное имя EXE-файла>”), что позволяет, например, управлять аппаратурой измерения и управления, соединенной с компьютером с помощью интерфейса, и вносить в базу данных данные, полученные с помощью этой аппаратуры.

## **4.2. Создание и сохранение файла базы данных**

- a) #Файл# - [Создать] – [Пустая база данных рабочего стола]
- b) на появившейся диалоговой панели щелкнуть левой кнопкой мыши на пиктограмме «Поиск расположения для размещения базы данных» и выбрать каталог, в котором необходимо поместить файл базы данных;
- c) внести имя файла базы данных в окно ввода и [Создать].

## **4.3. Создание таблицы**

Создание объектов базы данных производится с помощью средств управления (пиктограмм), размещенных на вкладке «Создание». Вид окна разработки объектов Access 2016 приведен на рис. 4.3.

Создание таблицы производится в режиме КОНСТРУКТОР. В этом режиме задаются названия полей таблицы, типы и формат данных, назначается ключевое поле.

В окне базы данных (рис. 4.3) выполнить действия:

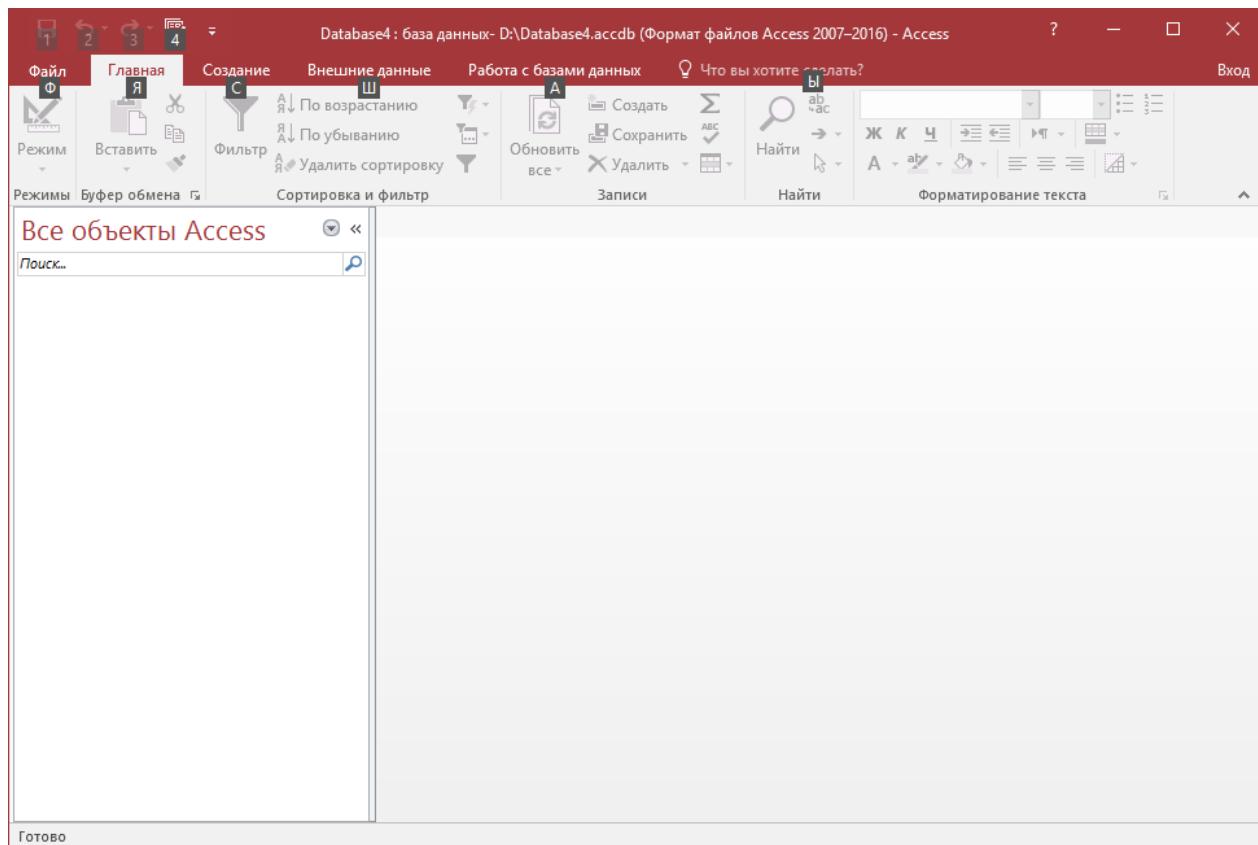


Рис. 6.

- a) #Создание# – !Таблицы! - [Конструктор таблиц];
- b) заполнить в режиме «Конструктор» поля таблицы (см. пример на рис. 4.4): «Имя поля», «Тип данных» и «Свойства поля»;
- c) сохранить макет таблицы:  
    #Файл# - [Сохранить как] – [Сохранит объект как...] - {имя таблицы}.
- d) переключиться в режим таблицы:  
    #Конструктор# - !Режимы! – [Режим] - [Режим таблицы];
- e) заполнить таблицу. Хотя бы одно из полей таблицы должно быть назначено как ключевое. Ключевое поле должно однозначно определять набор записей в таблице. Ключевым полем в таблице, содержащей сведения о сотрудниках или студентах, может быть номер паспорта или номер зачетной книжки.  
Если поля, которое однозначно определяло бы набор записей, в таблице нет, то оно создается искусственно с типом данных – счетчик (Autonumber). Для того, чтобы назначить поле ключевым, нужно пометить поле (строку), вызвать контекстное меню и выбрать в нем пункт «Ключевое поле»;

- f) сохранить таблицу:  
 #Файл# - [Сохранить] .

В дальнейшем при необходимости редактирования содержимого таблицы переходить в «Режим таблицы», при необходимости редактирования заголовков столбцов таблицы, типов данных и свойств полей - переходить в режим «Конструктор».

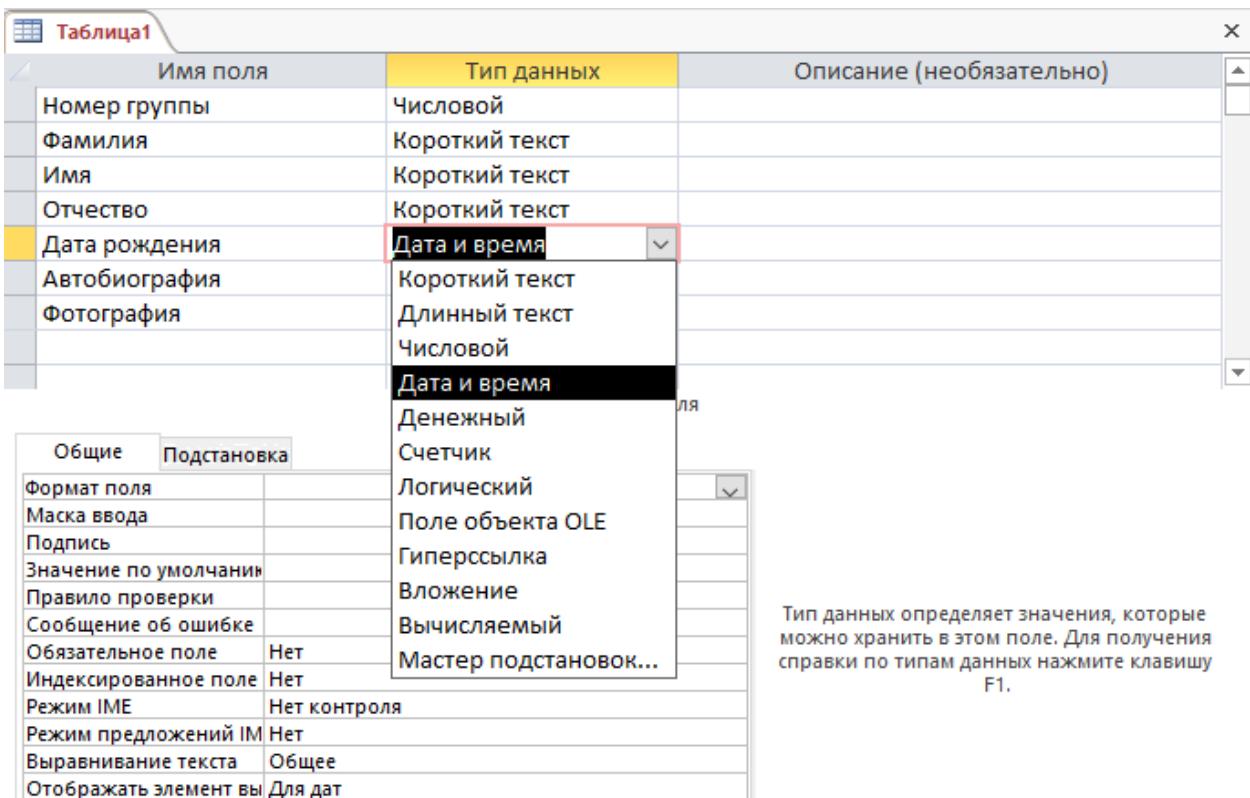


Рис. 4.4.

#### 4.4. Создание и использование форм

Формы делают более удобным доступ к записям (строкам) таблицы (нескольких таблиц) при просмотре, редактировании, введении новых записей и удалении записей. Специальный вид формы – кнопочная форма, которая содержит только средства доступа (кнопки) к объектам базы данных.

##### Создание формы

- а) #Создание# - !Формы! –[Мастер форм];

- b) во всплывающей панели (рис. 4.5) выбрать поля таблицы, для которых создается пользовательская форма (можно выбрать все поля, можно не все);
- c) выполнить действия по запросам Мастера форм.

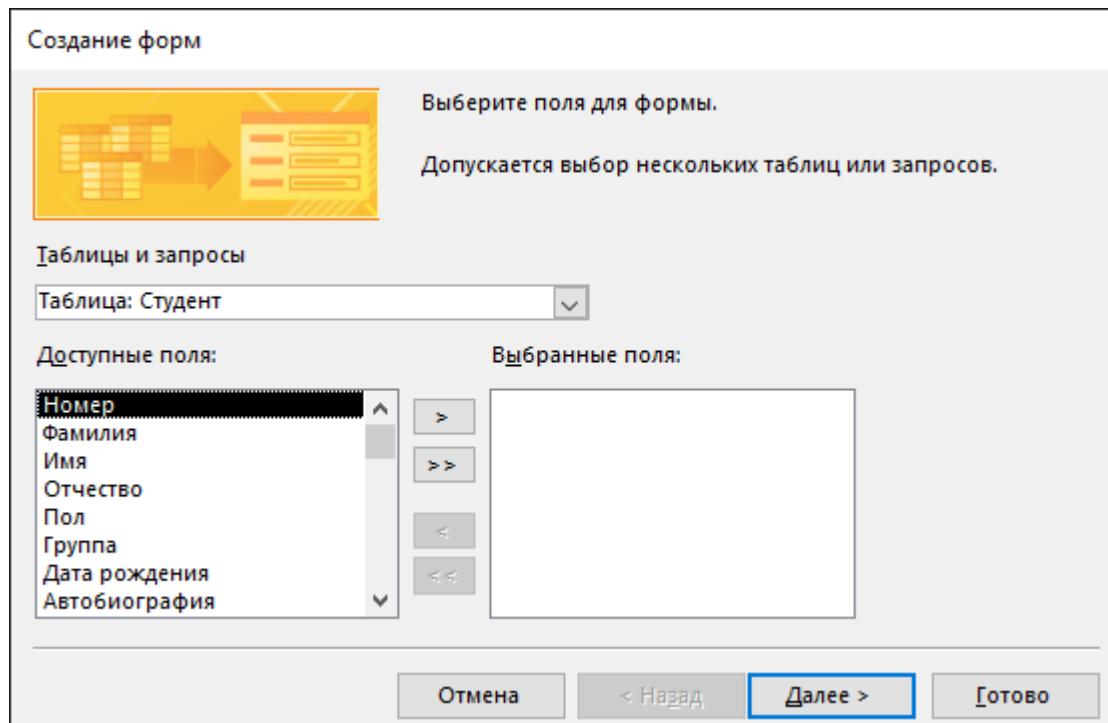


Рис. 4.5.

### **Поиск данных в формах**

- a) пометить поле формы, в котором производится поиск;
- b) #Главная# - [Найти] – [Найти] - {образец} - [Найти далее].

### **Сортировка данных (по возрастанию/убыванию)**

#Главная# - !Сортировка и фильтр! – [по возрастанию]  
- [по убыванию].

### **Фильтрация данных (фильтр по выделенному)**

- a) пометить в таблице/форме признак поиска в записи интересующего поля (например, год рождения, пол);

b) #Главная# - !Сортировка и фильтр! – [Выделение].

Для отмены фильтрации:

#Главная# - !Сортировка и фильтр! – [Удалить фильтр].

### **Фильтрация данных (фильтр по форме)**

a) #Главная# - !Сортировка и фильтр! - [Фильтр] - [Текстовые фильтры];

b) ввести признак поиска (равно, не равно, начинается с... и т.д.);

Для отмены фильтрации:

#Главная# - !Сортировка и фильтр! – [Удалить фильтр].

### **Фильтрация и сортировка данных**

#Главная# - !Сортировка и фильтр! - [Расширенный фильтр];

задать условия сортировки и фильтрации;

#Главная# - !Сортировка и фильтр! - [Применить фильтр].

### **Создание запросов к однотабличной базе данных**

#Создание# - !Запросы! - [Мастер запросов] – для создания запроса с

помощью Мастера;

#Создание# - !Запросы! - [Конструктор запросов] – для создания запроса в режиме «Конструктор».

### **4.5. Создание реляционной БД и работа с ней**

Реляционная БД представляет совокупность связанных таблиц. Понятие "связанных" означает, что в таблицах есть связанные поля. С одной строкой одной таблицы может быть связана только одна строка другой (тип связи: один-к-одному), или несколько (тип связи: один-ко-многим). Если с одной

строкой одной связано несколько строк другой и с одной строкой другой связано несколько строк одной то тип связи будет многие-ко-многим.<sup>14</sup>

## Формирование структуры связей между таблицами

При построении схемы данных Access автоматически определяет тип связи 1:1 или 1:M, при условии, если поле, по которому нужно установить связь, является ключевым в главной и/или подчиненной таблице.

Если поле, по которому устанавливается связь, является уникальным ключом и в главной и в подчиненных таблицах - автоматически устанавливается тип связи 1:1.

Если поле, по которому устанавливается связь, является уникальным ключом в главной, а в подчиненной таблице не является ключевым или входит в составной ключ - автоматически устанавливается тип связи 1:M.

При выборе в качестве поля связи в главной таблице **неключевого** поля тип отношения автоматически не определяется. В этом случае между таблицами устанавливаются связи-объединения.

Связи-объединения возможны трех видов:

1. Объединение только тех записей, в которых связанные поля обеих таблиц совпадают.
2. Объединение тех записей, в которых связанные поля обеих таблиц совпадают, а также объединение ВСЕХ записей из первой таблицы, для которых нет связанных во второй, с пустой записью второй таблицы; (Объединение ВСЕХ записей первой таблицы и только тех из второй, в которых связанные поля совпадают).
3. Объединение тех записей, в которых связанные поля обеих таблиц совпадают, а также объединение ВСЕХ записей из второй таблицы, для которых нет связанных в первой, с пустой записью первой таблицы (Объединение ВСЕХ записей из второй таблицы и только тех из первой, в которых связанные поля совпадают).

---

<sup>14</sup> В случае связи многие-ко-многим для возможности обеспечения целостности данных необходимо создавать дополнительную таблицу (см. упражнения, таблицу Книги/Авторы). В результате вместо одной связи многие-ко-многим образуются две связи один-ко-многим.

Тип объединения должен (может) устанавливать пользователь. По умолчанию устанавливается первый тип.

Для связей-объединений условие «Обеспечение целостности данных» не устанавливается. Практически это означает, что если в одну из таблиц будут вноситься изменения, то на содержании второй это не отразится. За целостностью данных должен следить пользователь!

## **Обеспечение целостности данных в БД**

Параметр «Обеспечение целостности данных», а также «Каскадное обновление связанных записей» и «Каскадное удаление связанных записей» можно установить для связей 1:1 и 1:М.

Обеспечение целостности данных означает выполнение для взаимосвязанных таблиц следующих условий корректировки БД:

- в подчиненную таблицу не может быть добавлена запись с не существующим в главной таблице значением ключевого поля;
- в главной таблице нельзя удалить запись, если не удалены связанные с ней записи в подчиненной таблице;
- изменение значений ключа связи главной таблицы должны приводить к изменению соответствующих значений в записях подчиненной таблицы.

Если установлен только параметр «Обеспечение целостности данных», то при попытке нарушить это условие Access выдает предупреждение. Если установлены параметры каскадного обновления и удаления записей, то Access будет автоматически производить корректировку данных в связанных таблицах.

## **Нормализация таблиц.**

Наличие в каком-либо одном или в нескольких столбцах таблицы повторяющихся записей, например, названия кафедры, номера телефона и т.д. является признаком того, что требования нормализации для таблицы не выполнены, что имеется не устраниенная транзитивная зависимость и что,

следовательно, необходимо разделить эту таблицу на две или более связанных таблиц.

### Роль связей между таблицами. Установление целостности данных.

Связи между таблицами нужны для:

- автоматического контроля средствами Access за целостностью данных;
- возможности использования средств автоматического конструирования запросов и отчетов.

Пример. База данных "Обучение". Схема данных может иметь вид рис. 4.6.

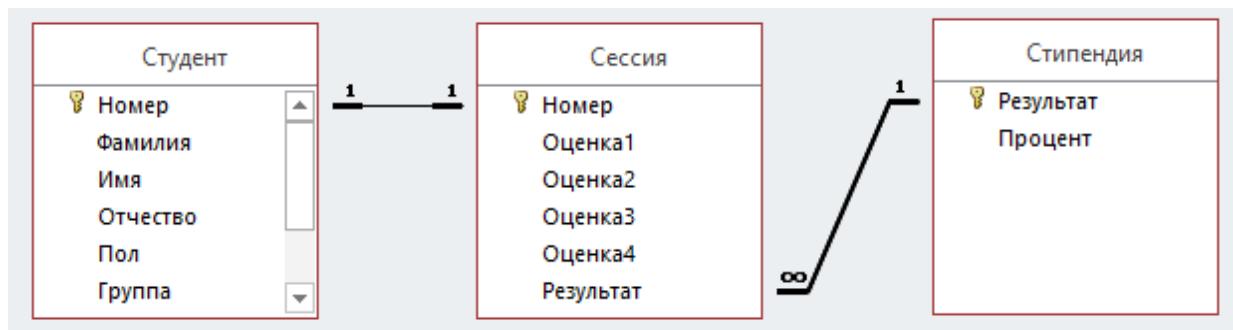


Рис. 4.6.

Связь всегда устанавливается между двумя таблицами. Для любой связи одна таблица является главной, другая подчиненной. В паре СТУДЕНТ-СЕССИЯ связь типа 1:1, поэтому формально главной может быть как СТУДЕНТ так и СЕССИЯ. Однако, правильно сделать главной СТУДЕНТ, т.к. по правилам обеспечения целостности данных в главной таблице должно быть столько же или больше записей, чем в подчиненной. Сведения о студентах, безусловно, первичны, т.к. в сессию сдать экзамены могут и не все студенты, которые есть в списке (в таблице СТУДЕНТ).

В паре СЕССИЯ-СТИПЕНДИЯ главной должна быть таблица СТИПЕНДИЯ по двум причинам:

- в таблице СТИПЕНДИЯ поле Результат - ключевое, а в таблице СЕССИЯ поле Результат - не является ключевым.
- таблица СТИПЕНДИЯ является справочной и в ней предусмотрены все возможные варианты оценок, а в таблице СЕССИЯ встретиться могут и не

все оценки (например, все студенты сдадут экзамены на 5), т.е. в таблице СТИПЕНДИЯ по параметру «Результат» всегда столько же или больше вариантов, чем в таблице СЕССИЯ.

### *Как установить подчиненность таблиц?*

В процессе установления связей (#Работа с базами данных# – [Схема данных...]) можно задать не только саму связь, но и подчиненность. Главной таблицей будет та, от ключевого поля которой будет прокладываться линия связи. Как задать подчиненность - должен решить пользователь!

### *В чем выражается на практике обеспечение целостности данных?*

а) в подчиненную таблицу не может быть добавлена запись с не существующим в главной таблице значением ключа связи.

Пример 1. В БД "Обучение" в таблицу СЕССИЯ нельзя будет ввести оценки студента с номером зачетной книжки (ключевое поле), которого нет в таблице СТУДЕНТ. (А в таблицу СТУДЕНТ - ввести новую запись с данными нового студента - можно!)

б) в главной таблице нельзя удалить запись, если не удалены связанные с ней записи в подчиненной таблице.

Пример 2. Если в БД "Обучение" делается попытка удалить строку из таблицы СТУДЕНТ, то автоматически будут удалена и запись оценок в сессию для этого студента в таблице СЕССИЯ. (Но не наоборот, т.к. главная таблица - СТУДЕНТ, а таблица СЕССИЯ - подчиненная).

Пример 3. В таблице СТИПЕНДИЯ нельзя удалить ни одну из строк: нхр, отл, хор, хр1, т.к. при таком удалении автоматически будут удалены соответствующие строки таблицы СЕССИЯ, имеющие то же значение в столбце Результат. (Но не наоборот: удалить любую строку в таблице СЕССИЯ можно без всяких последствий для таблицы СТИПЕНДИЯ).

## **4.6. Технология загрузки таблиц БД с использованием форм**

Данные в таблицы БД должны загружаться в следующем порядке:

1. Независимо могут загружаться таблицы, которые не подчинены каким-либо другим таблицам в одно-многозначных связях.

2. Таблицы, подчиненные каким-либо другим, могут загружаться либо одновременно с ними, либо после загрузки главных таблиц. В противном случае не смогут установиться связи загружаемых в подчиненную таблицу записей с записями в главных таблицах.

3. В базу данных сначала загружаются с соответствующих документов справочные данные, а затем учетные.

#### **4.7. Создание реляционной БД**

- a) #Файл# - [Создать] – [Новая база данных]
- b) на появившейся диалоговой панели щелкнуть левой кнопкой мыши на пиктограмме «Поиск расположения для размещения базы данных» и выбрать каталог, в котором необходимо поместить файл базы данных;
- c) внести имя файла базы данных в окно ввода и [Создать].

#### **Создание таблиц БД**

Создать таблицы. Для этого:

- a) #Создание# - !Таблицы! - [Конструктор таблиц];
- b) заполнить в режиме «Конструктор» поля таблицы (см. пример на рис. 4.4): «Имя поля», «Тип данных» и «Свойства поля»;
- c) задать в столбце «Имя поля» имена полей в таблице;
- d) задать в столбце «Тип данных» типы данных для всех полей. Для этого: активизировать клетку столбца «Тип данных», в результате в клетке появится кнопка вызова меню; открыть меню и выбрать необходимый тип<sup>15</sup> поля; откорректировать при необходимости параметры формата (например, максимальное количество символов при текстовом формате) во вкладке «Свойства поля».
- e) установить признак ключевого поля в одной из строк. Для этого: пометить строку; вызвать контекстное меню щелчком правой кнопки мыши; выбрать в контекстном меню пункт «Ключевое поле»;
- f) сохранить таблицу:

---

<sup>15</sup> Кроме обычных типов поля (Текстовое, Числовое и т.д.) предоставляются Поле Длинный текст - для ввода значительных фрагментов текста; Поле объектов OLE - для ввода объектов OLE, например, рисунков из графических файлов

#Файл# – [Сохранить] – {имя таблицы}

### Формирование структуры связей между таблицами

- #Работа с базами данных# - [Схема данных];
- вызвать контекстное меню во всплывающей панели «Схема данных» и выбрать пункт «Добавить таблицу...»;
- ввести последовательно таблицы во всплывающую панель «Схема данных» из окна со списком вкладки «Таблицы» всплывающей панели «Добавление таблицы»:  
[имя таблицы] – [Добавить] ;
- закрыть панель ввода «Добавление таблицы»:  
[Закрыть];
- установить связи между полями таблиц путем буксировки «Катушки с проводом» от поля таблицы-источника к полю таблицы приемника;
- во всплывающей панели «Изменение связей» рис. 4.7 для связей типа «один-к-одному» и «один-ко-многим» установить опции:

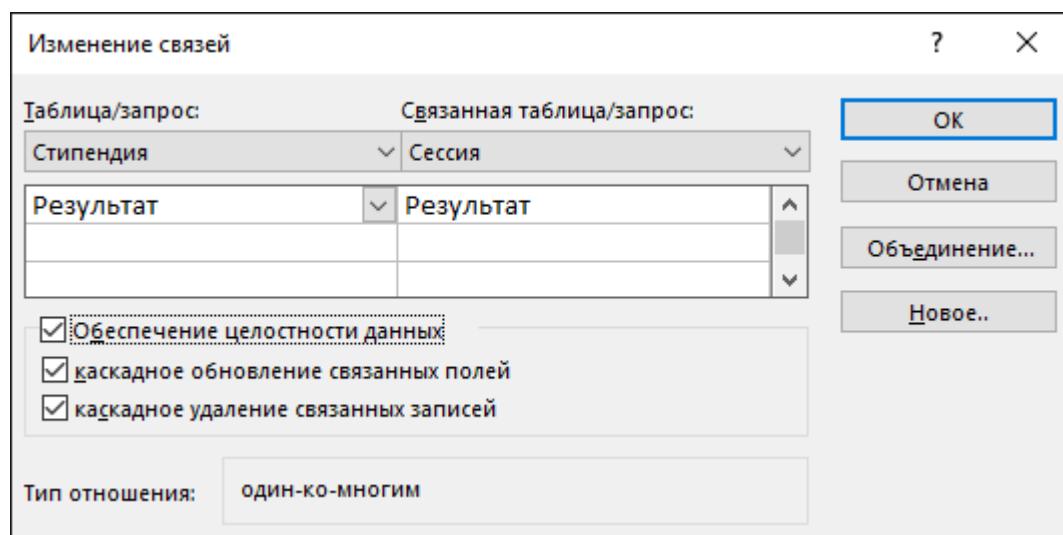


Рис. 4.7.

- обеспечение целостности данных;
- каскадное обновление связанных записей;
- каскадное удаление связанных записей.

Окончание установок – [Создать].

Для изменения типа связи нужно:

- промаркировать связь;

- b) вызвать контекстное меню, пункт «Изменить связь...»;
- c) во всплывающей панели «Изменение связей» (рис. 47) выбрать необходимый тип связи.

## **Создание форм**

Формы создаются для удобства заполнения и редактирования данных в таблицах. Количество форм может быть меньше, чем число таблиц. В таблицы, для которых формы не созданы, данные вводятся непосредственно<sup>16</sup>. С помощью одной формы можно вводить и редактировать данные в двух и более таблицах.

Для создания формы нужно выполнить следующую последовательность действий:

- a) #Создание# - !Формы! –[Мастер форм];
- b) в окне «Таблицы и запросы» всплывающей панели «Создание форм» выбрать имя таблицы или запроса, для которой создается форма;
- c) выбрать из меню поля одной (основной) таблицы/запроса и, если требуется, других, которые нужно включить в форму. Для этого использовать кнопку ">", если необходимо включить в форму какое-либо одно поле таблицы, и кнопку ">>", если нужно включить в форму все поля таблицы;
- d) следуя указаниям Мастера форм, довести процесс создания формы до завершения, затем закрыть панель «Создание форм» .

## **Создание поля со списком в формах.**

Если в каком-либо поле основной таблицы, на базе которой создается форма, содержится не наименование, а код атрибута (этоично для БД), а наименование - в другой, связанной с ней справочной, то целесообразно создать в форме не текстовое поле, а поле со списком. В таком случае при работе с формой данные в таком окне не нужно будет вводить с клавиатуры, достаточно будет выбирать пункт из поля со списком. Естественно, что соответствующая связанная таблица должна быть предварительно создана и данные в нее должны быть введены.

---

<sup>16</sup> При вставке в таблицу объекта OLE можно сделать его встроенным в БД, или связанным. В последнем случае БД будет иметь меньший размер, а содержимое графического файла можно изменять, не затрагивая БД. По умолчанию объект OLE встраивается, для того, чтобы сделать его встроенным, нужно установить флажок "Связь" в процессе вставки объекта в таблицу.

Для создания поля со списком в окне формы вместо создающегося по умолчанию окна ввода (поля):

- a) перейти в режим «Конструктор»;
- b) активизировать окно ввода, вместо которого должно быть создано поле со списком, и удалить его;
- c) #Конструктор# - !Элементы управления! – [Поле со списком];
- d) создать вместо окна ввода, используя панель элементов, поле со списком. Для этого при нажатой левой кнопке мыши «нарисовать» контур окна ввода. Далее в ходе диалога с Мастером форм:
  - указать источник данных (справочную таблицу);
  - указать в справочной таблице поля с кодом атрибута и наименованием атрибута;
  - установить опцию «Скрыть ключевой столбец». В этом случае в окне формы будет отображаться только наименование атрибута, а код отображаться не будет;
  - установить опцию «Сохранить в поле» и выбрать в окне ввода из списка название поля, вместо которого создается поле со списком;
  - заменить стандартное название поля со списком на требуемое, отражающее содержание данных в поле со списком.
- e) сохранить изменения макета и закрыть окно формы.

### **Создание поля со списком в окнах подчиненных форм**

Если форма создается на базе основной и связанной таблицы (или запроса), то Мастер форм автоматически создаст основную и дополнительную подчиненную форму. Для создания поля со списком в окне подчиненной формы нужно «настроить» только дополнительную подчиненную форму, аналогично тому, как это описано в пункте «Создание поля со списком в окнах форм».

### **4.8. Создание отчетов**

Отчеты могут создаваться на основе данных из таблиц или запросов с помощью Мастера отчетов или в режиме «Конструктор».

1. Создание отчета с помощью Мастера отчетов.

- a) #Создание# - !Отчеты! – [Мастер отчетов]
  - b) по запросам Мастера отчетов выбрать поля из таблиц и запросов, которые должны войти в отчет, указать уровни группировки, порядок сортировки, итоги, вид макета отчета, стиль оформления;
  - c) доработать отчет в режиме «Конструктор» (Изменить названия, виды шрифта и т.д.), если это необходимо.
2. Создание отчета на основе запроса.
- a) создать запрос (см. раздел «Создание запроса»);
  - b) выполнить действия по п.1.

Построение отчета на основе запроса дает возможность включать в отчет не только первичные данные из таблиц, но и вычисляемые на их основе.

#### **4.9. Создание кнопочных форм управления в БД**

Кнопочные формы управления в Access дают возможность формировать средства графического пользовательского интерфейса - панель с кнопками, каждая из которых запускает определенное действие (открывает форму, отчет и т.д.)

- a) [Диспетчер кнопочных форм] на панели быстрого доступа<sup>17</sup> ;
- b) во всплывающем диалоговом окне «Диспетчер кнопочных форм» щелкнуть левой кнопкой мыши на кнопке “Изменить...”;
- c) во всплывающем диалоговом окне «Изменение страницы кнопочной формы» можно изменить название кнопочной формы, затем щелкнуть левой кнопкой мыши на кнопке “Создать...” и ввести элемент кнопочной формы для:
  - открытия формы в режиме добавления;
  - открытия формы в режиме редактирования;
  - открытия отчета;
  - выхода из приложения;
  - запуска макроса;
  - запуска программы.

---

<sup>17</sup> Если на панели быстрого доступа кнопки «Диспетчер кнопочных форм» нет, то ее нужно создать: ФАЙЛ/ПАРАМЕТРЫ/Панель быстрого доступа. Затем на вкладке «Настройка панели быстрого доступа» надо выбрать из раскрывающегося списка строку: Вкладка "Работа с базами данных"/ Диспетчер кнопочных форм/ДОБАВИТЬ.

- d) повторять п.п. б) и в) для последовательного ввода других элементов кнопочной формы;
- е) [Закрыть] – [Закрыть].

В результате будет автоматически создано два объекта базы данных: кнопочная форма и таблица «Switchboard Items», в которой находятся элементы кнопочной формы. Для удаления кнопочной формы и создания новой предварительно нужно удалить оба эти объекта.

#### **4.10. Параметры запуска**

Можно сделать так, чтобы при открытии БД автоматически производился вызов какой-либо формы, например, кнопочной формы. Для этого

- а) #Файл# – [Параметры] – [Текущая база данных];
- б) выбрать название формы в поле со списком «Форма просмотра» (например, «Кнопочная форма», если вызываться должна кнопочная форма).

Можно установить также элементы оформления: заголовок приложения и значок приложения. Они появятся в строке заголовка окна Access при вызове данной БД и отобразятся на пиктограмме на панели задач вместо стандартных. Для установки значка приложения нужно в окне ввода «Значок приложения» ввести имя и путь к файлу <имя>.ico, содержащему изображение значка.

### **Упражнения 1**

#### **Создание однотабличной базы данных (БД)**

Создание новой базы данных

- а) создайте новую базу данных sess.accdb;
- б) создайте в режиме «Конструктор» структуру таблицы СТУДЕНТ по образцу табл. 4.1;
- с) сохраните таблицу;
- д) заполните таблицу базы данных СТУДЕНТ по образцу табл. 21.

Табл. 4.1.

Имя поля	Тип данных	Размер и формат поля <sup>18</sup>
Номер	Числовой	6
Фамилия	Короткий текст	15
Имя	Короткий текст	10
Отчество	Короткий текст	15
Пол	Короткий текст	1
Дата рождения	Дата и время	Краткий формат даты
Группа	Короткий текст	3
Автобиография	Длинный текст	
Фотография	Поле объекта OLE	

Табл. 4.2.

Номер	Фамилия	Имя	Отчество	Пол	Дата рождения	Группа
16693	Анохин	Андрей	Борисович	М	24.02.95	10842
16593	Петрова	Анна	Владимировна	Ж	15.03.95	10842
16993	Кравцов	Алексей	Иванович	М	09.09.95	10842
16493	Сергеев	Петр	Михайлович	М	01.01.96	10841
16793	Борисова	Мария	Михайловна	Ж	14.04.96	10841
16893	Зайцев	Сергей	Александрович	М	29.07.96	10841
17093	Волкова	Светлана	Николаевна	Ж	07.12.96	10841
16333	Шарапова	Мария	Юрьевна	Ж	01.01.98	10843

*Указания по выполнению*

- а) внесите в поле "Автобиография" или перенесите через буфер обмена автобиографию из файла Шарапова.doc, для удобства просмотра вызовите нажатим клавиш <Shift+F2> увеличенное окно ввода;

---

<sup>18</sup> Размер и формат поля устанавливается на вкладке «Свойства поля»

- b) в поле "Фотография" вставьте объект из файла M\_SH.bmp. Для этого откройте файл в редакторе Paint и перенесите изображение через буфер обмена. Просмотр объекта после ввода - двойным щелчком мыши;
- c) сохраните таблицу.

### **Создание однотабличных пользовательских форм**

- a) создайте однотабличную пользовательскую форму для таблицы СТУДЕНТ;
- b) произведите просмотр и редактирование данных в таблице с использованием пользовательской формы;
- c) произведите подстройку размеров полей «Автобиография» и «Фотография» в режиме «Конструктор».

### **Поиск и отбор данных**

**1. Поиск данных.**

Произведите поиск записей в таблице (форме) СТУДЕНТ:

- по признаку - год рождения -1996;<sup>19</sup>
- по признаку - фамилия начинается с буквы С.<sup>20</sup>

**2. Сортировка данных.**

Отсортируйте записи в таблице СТУДЕНТ в соответствии с алфавитным порядком записей в столбце ФАМИЛИЯ.

**3. Фильтрация данных в таблицах и формах.**

- a) произведите фильтрацию данных в режиме «Фильтр по выделенному» по признаку: год рождения -96, затем дополнительно по признаку: пол – м; После просмотра результатов – восстановите исходный вид таблицы;
- b) найдите записи в режиме фильтрации «Расширенный фильтр» по условию: сортировка по фамилии, выборка по условию: год рождения <05.05.96 или группа - 10841.

**4. Создание запросов к однотабличной БД.**

- a) создайте простой запрос к таблице СТУДЕНТ на выборку полей: номер, фамилия, имя, отчество, пол, год рождения;

---

<sup>19</sup> Шаблон поиска ??????86 или \*96

<sup>20</sup> Шаблон поиска С\*

б) создайте в режиме «Конструктор» простой запрос к таблице СТУДЕНТ на выборку полей: номер, фамилия, имя, отчество, пол, год рождения с условием сортировки по столбцу «Фамилия» и отбора записей по условию «>01.01.96 And <30.12.99» в столбце «Дата рождения».

### **Вывод данных**

1. Вывод на печать таблиц и форм.

- а) выведите на печать таблицу СТУДЕНТ;
- б) выведите на печать форму СТУДЕНТ.

2. Создание отчета.

а) используя имеющуюся таблицу СТУДЕНТ, создайте и произведите предварительный просмотр перед печатью отчет в следующей форме:

СТУДЕНТ

Табл. 4.3

Фамилия	Имя	Отчество	Группа
Анохин	Андрей	Борисович	10842
Петрова	Анна	Владимировна	10841
Кравцов	Алексей	Иванович	10841
Сергеев	Петр	Михайлович	10841
Борисова	Мария	Михайловна	10842
Зайцев	Сергей	Александрович	10842
Волкова	Светлана	Николаевна	10843
Сатина	Анастасия	Сергеевна	10841

б) в режиме «Конструктор» измените название отчета на «Список студентов».

### **Упражнения 2**

#### **1. Создание реляционной БД "Обучение"**

а) введите в структуру таблицы СТУДЕНТ признак ключа в поле "Номер" и создайте таблицы СЕССИЯ и СТИПЕНДИЯ в соответствии со следующей структурой:

Табл. 4.4. Структура таблицы СТИПЕНДИЯ

	<b>Имя поля</b>	<b>Тип данных</b>	<b>Размер поля</b>	<b>Формат поля</b>
Ключевое поле	Результат	Короткий текст	3	
	Процент	Числовой	Одинарное плавающей точкой	с Процентный

Табл. 4.5. Структура таблицы СЕССИЯ

	<b>Имя поля</b>	<b>Тип данных</b>	<b>Размер поля</b>	<b>Формат</b>
Ключевое поле	Номер	Короткий текст	5	
	Оценка1	Числовой	Длинное целое	Фиксированный
	Оценка2	Числовой	Длинное целое	Фиксированный
	Оценка3	Числовой	Длинное целое	Фиксированный
	Оценка4	Числовой	Длинное целое	Фиксированный
	Результат	Короткий текст	3	

### Установление связей между таблицами БД "Обучение"

Установите связи между полями таблиц СТИПЕНДИЯ/Результат - СЕССИЯ/Результат и СЕССИЯ/Номер - СТУДЕНТ/Номер.

### Создание форм для работы с таблицами

Заполните таблицы СТИПЕНДИЯ и СЕССИЯ по образцам табл.4.6, 4.7:

Табл. 4.6.

	Результат	Процент
[+]	нхр	0,00%
[+]	отл	200,00%
[+]	хор	150,00%
[+]	xp1	100,00%

Табл. 4.7.

	Номер	Оценка1	Оценка2	Оценка3	Оценка4	Результат
[+]	16333	5	5	5	5	отл
[+]	16493	3	3	3	3	нхр
[+]	16593	5	5	4	5	хор
[+]	16693	5	5	5	5	отл
[+]	16793	4	4	5	5	хр1

### Формирование отчетов

- a) сформируйте отчет «Проект приказа» со структурой: группа, фамилия, имя, отчество, процент;
- b) сформируйте отчет «Студент» в соответствии с одной из структур:
  - номер (зачетной книжки), фамилия, имя, отчество, пол, дата рождения, группа;
  - фамилия, имя, отчество, группа, автобиография;
  - фамилия, имя, отчество, фотография.

### Создание кнопочных форм для БД "Обучение"

Создайте главную кнопочную форму для БД с функциями:

- ввод/просмотр формы СТУДЕНТ;
- ввод/просмотр формы СЕССИЯ;
- просмотр отчета ПРОЕКТ ПРИКАЗА;
- ВЫХОД.

### Установка параметров запуска для БД "Обучение"

Установите параметры запуска для автоматического открытия главной кнопочной формы при открытии БД.

## 2. Создание реляционной БД "Библиотека" и средств пользовательского интерфейса (форм, запросов, отчетов) для работы с ней

2.1 Создайте в режиме «Конструктор» таблицы: АВТОРЫ, КНИГИ, КНИГИ/АВТОРЫ, РАЗДЕЛЫ и ЦИТАТЫ базы данных "Библиотека".

2.2 Сформируйте структуру связей между таблицами БД "Библиотека" по приведенной схеме рис. 4.8.

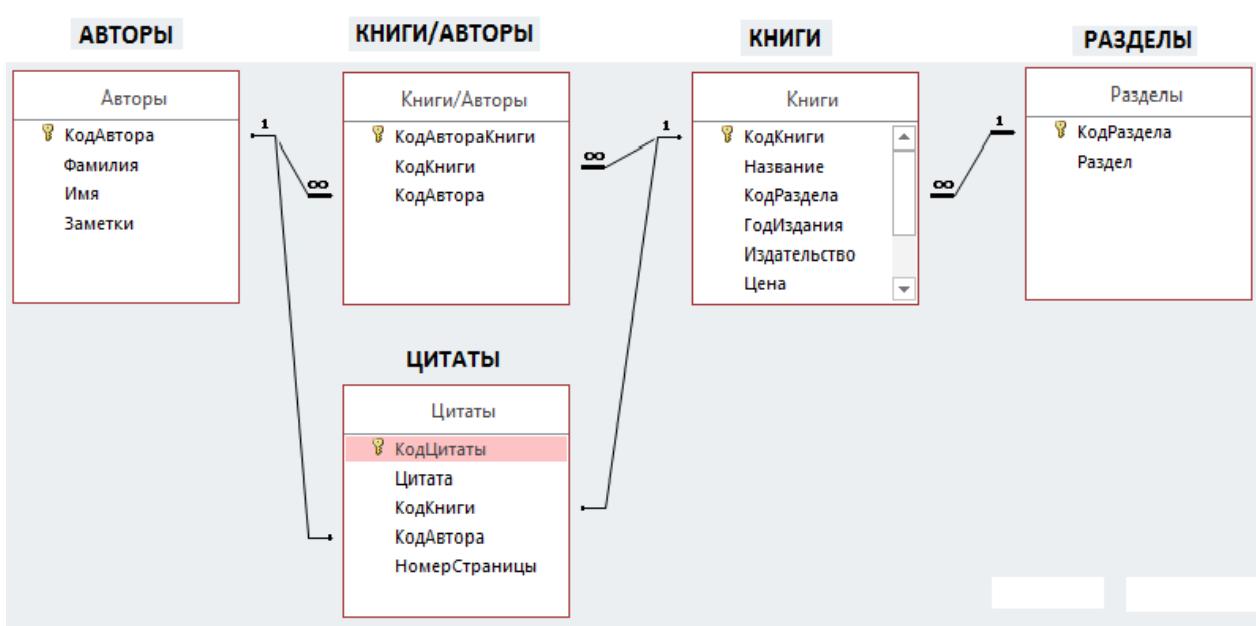


Рис. 4.8.

2.3. Создайте формы АВТОРЫ, РАЗДЕЛЫ, ЦИТАТЫ и заполните, используя формы, таблицы АВТОРЫ и РАЗДЕЛЫ, по данным в файле literary.docx.

2.4. Создайте форму КНИГИ, включив в нее все поля таблицы КНИГИ и поле "Код автора" из таблицы КНИГИ/АВТОРЫ (см. рис. 4.9).

2.5. Отредактируйте форму КНИГИ в режиме «Конструктор». Создайте поле со списком вместо окна «Код раздела», чтобы отображались только наименования разделов: Учебная, Художественная, Справочники и т.д.

2.6. Замените в режиме «Конструктор» поле «Код автора» на поле со списком в подчиненной форме КНИГИ/АВТОРЫ, настроив его на отображение фамилии (или фамилии и имени) автора.

Книги

КодКниги	2	ДатаПокупки				
Название	Почему Россия не Америка	ЧислоСтраниц	410			
Раздел	Публицистика	Заметки	Книга для тех, кто остается в России			
ГодИздания	2001					
Издательство	Крымский мост - 9Д					
Цена	100,00 ₽					
Книги/Авторы <table border="1"> <thead> <tr> <th>Автор</th> </tr> </thead> <tbody> <tr> <td>Паршев</td> </tr> <tr> <td>*</td> </tr> </tbody> </table>				Автор	Паршев	*
Автор						
Паршев						
*						

ОткроФРазделы

Рис. 4.9.

2.7. Замените в режиме «Конструктор» поля «Код автора» и «Код книги» формы ЦИТАТЫ на поля со списком, настроив их на отображение фамилии автора и названия книги соответственно. Заполните с помощью созданной формы по данным в файле literary.docx таблицу ЦИТАТЫ.

2.8. Создайте отчет «Авторский каталог», в который включите поля: «Фамилия», «Имя» из таблицы АВТОРЫ, «ОбычнНазвание», «Заметки» из таблицы КНИГИ и поле «Раздел» из таблицы РАЗДЕЛЫ. Отредактируйте отчет в режиме «Конструктор»: отрегулируйте размеры и наименования полей, измените название отчета.

2.9. Создайте кнопочную форму, в которую включите:

- вызов формы АВТОРЫ;
- вызов формы КНИГИ;
- просмотр отчета АВТОРСКИЙ КАТАЛОГ;
- выход.

В качестве элемента оформления панели "Кнопочная форма" используйте рисунок из файла books.bmp.

2.10. Установите параметры запуска для автоматического открытия главной кнопочной формы при открытии БД.

### **Указания по выполнению п. 2.3**

При создании поля со списком «Код раздела» в форме КНИГИ на панели свойств созданного поля со списком в строке «Данные» вкладки «Данные» выберите «Код раздела».

### **Указания по выполнению п. 2.4**

При настройке подчиненной формы КНИГИ/АВТОРЫ на панели свойств созданного поля со списком:

- в строке «Данные» вкладки «Данные» выберите "Код автора";
- убедитесь, что в строке «Источник строк» записан текст запроса на языке SQL:

```
SELECT DISTINCTROW [Авторы].[КодАвтора],[Авторы].[Фамилия] FROM
[Авторы];
```

Для того, чтобы в подчиненной форме отображалась не только фамилия, но и имя автора, вызовите построитель запросов и создайте интерактивно в строке «Источник строк» следующую запись:

Поле	КодАвтора	[Фамилия]&" "&[Имя]
Имя таблицы	Авторы	

### **4.12. Макросы**

Макросы используются для автоматизации исполнения последовательности рутинных операций с БД, таких как:

- a) автоматическое открытие и/или печать нескольких форм или отчетов;
- b) автоматическое экспортование каких-либо данных из БД в другой формат (например, в dBase).

Макрос представляет собой одну или последовательность нескольких макрокоманд. Список основных макрокоманд приведен в табл. 4.8. Список макрокоманд может быть увеличен. Для этого:

#Конструктор# – !Показать или скрыть! – [Показать все действия]

Табл. 4.8. Основные макрокоманды.

ВосстановитьОкно	ОткрытьОтчет
ВывестиВформате	ОткрытьЗапись
ВыбратьОбъект	ОстановитьВсеМакросы
ВыделитьОбъект	Остановить Макрос
ВыполнитьКоманду	ВыйтиИзAccess
ДобавитьМеню	ОтменитьСобытие
ЗадатьЗначение	ОтправитьОбъектБД
ЗадатьКомандуМеню	ПерейтиК
ЗадатьФильтр	ПечатьОбъекта
ЗакрытьБазуДанных	ПесочныеЧасы
ЗакрытьОкно	ПоискЗаписи
ЗапускМакроса	ПоказатьВсеЗаписи
ЗапускПрограммы	ПредварительныйПросмотр
ЗапускКомандыМеню	ПрименитьФильтр
КопироватьОбъект	ПриОшибке
КэлементуУправления	ПрименитьФильтр
НаЗапись	РазвернутьОкно
НайтиЗапись	СвернутьОкно
Обновить	Сигнал
ОбновитьОбъект	СохранитьЗапись
Обновление	УдалитьЗапись
ОстановитьВсеМакросы	УдалитьФильтр
ОткрытьТаблицу	УдалитьВремПеременную
ОткрытьФорму	УдалитьВсеВремПеременные
ОткрытьЗапрос	УстраниТЬОшибкуМакроса

Макрокоманды имеют аргументы. Так например, макрокоманда «ОткрытьЗапрос» имеет аргумент Имя запроса;

Запуск макроса может осуществляться:

- автоматически при открытии БД;
- при активизации специально созданной для запуска макроса кнопке управления. Кнопка может быть создана в форме или отчете;

- c) автоматически при возникновении определенного события в форме, отчете или в элементе управления типа поле или поле со списком.

## **Создание макроса**

- a) #Создание# – !Макросы и код! - [Макрос];
- b) выбрать макрокоманду или последовательность макрокоманд из поля со списком в столбце «Добавить новую макрокоманду», установить при необходимости аргументы макрокоманд;
- c) сохранить макрос. Для этого:
  - сделать щелчок левой кнопкой мыши на пиктограмме «Сохранить» на панели быстрого доступа;
  - ввести имя макроса во всплывающее диалоговое окно «Сохранение».

## **Установка автоматического запуска макроса при открытии БД**

Для этого созданный макрос нужно сохранить с именем AutoExec.

## **Создание кнопки для запуска макроса**

Кнопки для запуска макроса могут быть созданы в формах и отчетах. Для создания кнопки:

- a) открыть форму или отчет в режиме «Конструктор»;
- b) используя элемент «Кнопка» панели элементов, создать кнопку в свободном поле формы или отчета;
- c) в процессе последующего, происходящего под управлением Мастера, задания свойств кнопки выбрать:  
[Разное] - [Выполнить макрос]  
и выбрать конкретный макрос (макрос должен быть заранее создан);
- d) сохранить форму (отчет) с кнопкой. Для этого сделать щелчок левой кнопкой мыши на пиктограмме «Сохранить» на панели быстрого доступа.

## **Установка режима запуска при возникновении события**

Событием, возникновение которого может приводить к автоматическому запуску макроса, может быть:

### ***а) в форме***

Текущая запись	До вставки	После вставки
До обновления	После обновления	Удаление
До подтверждения Del	После подтверждения Del	Открытие
Загрузка	Выгрузка	
Получение фокуса	Потеря фокуса	
Двойное нажатие кнопки	Кнопка вверх	Кнопка вниз
Перемещение указателя		
Нажатие клавиши	Перехват нажатия клавиши	
Клавиша вверх	Клавиша вниз	
Фильтрация	Применение фильтра	
Таймер	Интервал таймера	

### ***б) в отчете***

Открытие	Закрытие	Включение
Отключение	Отсутствие данных	Страница
Ошибка		

### ***в) в поле (поле со списком) формы или отчета***

До обновления	После обновления	Изменение
Вход	Выход	Получение фокуса
Потеря фокуса	(Отсутствие в поле со списком)	
Двойное нажатие кнопки	Кнопка вверх	Кнопка вниз
Перемещение указателя	Клавиша вниз	Клавиша вниз

Полный список событий в формах, отчетах, полях и полях со списком представлен на вкладках «Свойства» в окнах свойств соответствующих объектов.

Для установки события, в результате возникновения которого должен быть запущен макрос, нужно:

- a) открыть форму или отчет в режиме «Конструктор»;
- b) вызвать контекстное меню объекта-источника (если это форма - то в области «Примечание формы», если это поле - то в области поля), выбрать в меню пункт «Свойства»;
- c) в открывшейся панели настройки на вкладке «События» выбрать строку с названием события (например, «После обновления» и установить в ней имя макроса.

## **Создание и использование макросов с условием**

Если требуется, чтобы макрокоманды выполнялись только при соблюдении определенных условий, используйте блок **Если**. Он заменяет столбец Условие, который использовался в предыдущих версиях Access. Вы можете дополнить блок **Если** блоками **Иначе если** или **Иначе**, как и в других последовательных языках программирования, например VBA.

На следующем рисунке показан простой блок **Если**, включающий блоки **Иначе если** и **Иначе**:

Блок **Если** выполняется, если дата в поле ExpirationDate меньше текущей даты.

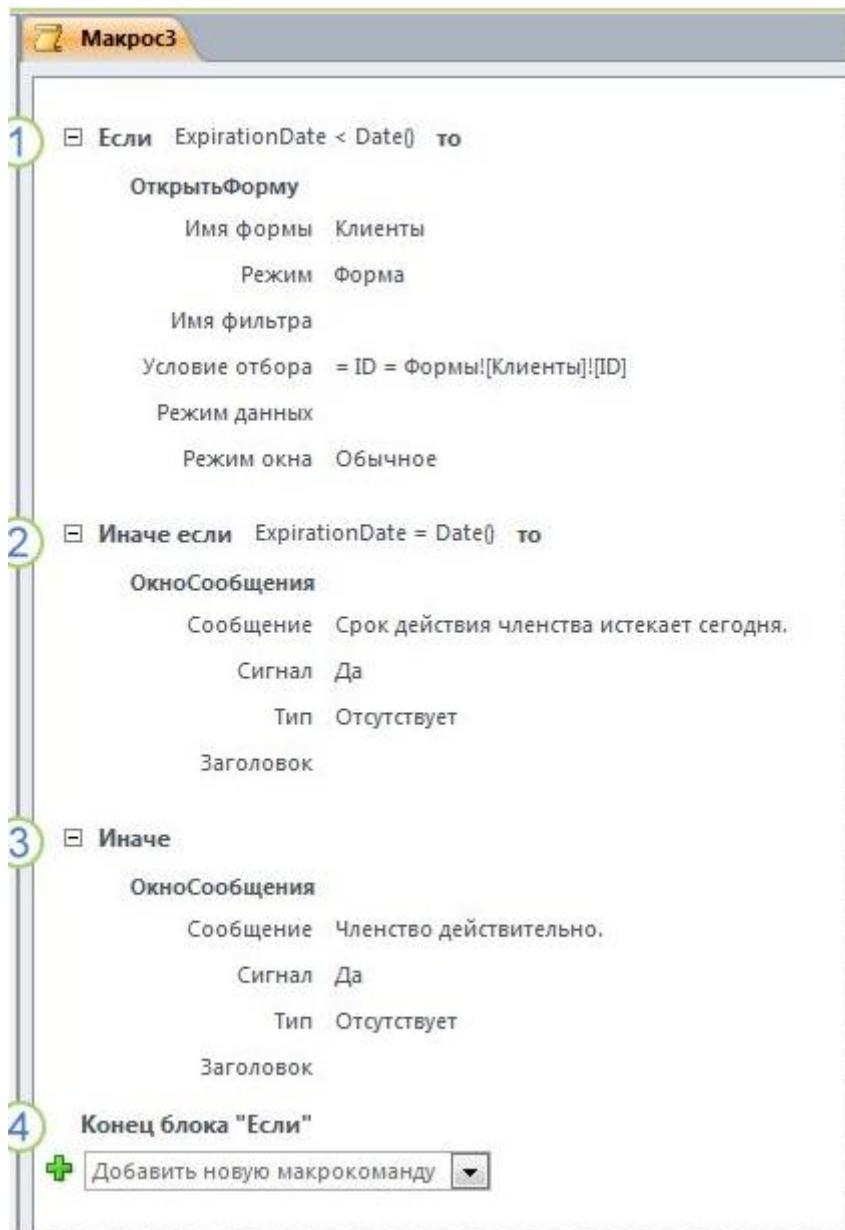
Блок **Иначе если** выполняется, если дата в поле ExpirationDate равна текущей дате.

Блок **Иначе** выполняется, если ни один из предыдущих блоков не выполнился.

На этом блок **Если** заканчивается.

### ***Добавление блока "Если" в макрос***

1. Выберите оператор **Если** в раскрывающемся списке **Добавить новую макрокоманду** или перетащите его из области "Каталог макрокоманд" в область макроса.
2. В поле в верхней части блока **Если** введите выражение, определяющее, когда будет выполняться блок. Выражение должно быть логическим (то есть его результатом должно быть значение "Да" или "Нет").
3. Добавьте макрокоманды в блок **Если**, выбрав их из раскрывающегося списка **Добавить новую макрокоманду** в блоке или перетащив их из области "Каталог макрокоманд" в блок **Если**.



### *Управление выполнением программы с помощью операторов "Если", "Иначе если" и "Иначе"*

Если требуется, чтобы макрокоманды выполнялись только при соблюдении определенных условий, используйте блок **Если**. Он заменяет столбец **Условие**, который использовался в предыдущих версиях Access. Вы можете дополнить блок **Если** блоками **Иначе если** или **Иначе**, как и в других последовательных языках программирования, например VBA.

На следующем рисунке показан простой блок **Если**, включающий блоки **Иначе если** и **Иначе**:

Блок **Если** выполняется, если дата в поле ExpirationDate меньше текущей даты.

Блок **Иначе если** выполняется, если дата в поле ExpirationDate равна текущей дате.

Блок **Иначе** выполняется, если ни один из предыдущих блоков не выполнился.

На этом блок **Если** заканчивается.

### ***Добавление блоков "Иначе" и "Иначе если" в блок "Если"***

1. Выберите блок **Если** и в его правом нижнем углу щелкните **Добавить блок "Иначе"** или **Добавить блок "Иначе если"**.
2. Если вы добавляете блок **Иначе если**, введите выражение, определяющее, когда будет выполняться блок. Выражение должно быть логическим (то есть его результатом должно быть значение "Истина" или "Ложь").
3. Добавьте макрокоманды в блок **Иначе если** или **Иначе**, выбрав их из раскрывающегося списка **Добавить новую макрокоманду** в блоке или перетащив их из области "Каталог макрокоманд" в блок.

#### **ПРИМЕЧАНИЯ :**

- Команды для добавления блоков **Если**, **Иначе если** и **Иначе** также доступны в контекстном меню, которое появляется при щелчке макрокоманды правой кнопкой мыши.
- У блоков **Если** может быть до 10 уровней вложенности.

### **Упражнения 3**

**Задание 1.** Создайте макросы для открытия и закрытия форм КНИГИ, АВТОРЫ, РАЗДЕЛЫ и ЦИТАТЫ и организуйте запуск каждого из них по одному из событий в форме: «Двойное нажатие кнопки», «Открытие», «Закрытие».

**Задание 2.** Автоматизируйте с помощью макросов процесс внесения дополнений в БД БИБЛИОТЕКА (library5.accdb) по следующему алгоритму:

- 1) открыть форму КНИГИ (при открытии БД);

2) вносить данные о книге. Если при заполнении поля со списком «Раздел» нужного раздела в списке нет - автоматически открывать форму РАЗДЕЛЫ, если есть - переходить к п.5;

3) после открытия формы РАЗДЕЛЫ закрыть форму КНИГИ (это необходимо для возможности автоматического обновления списка в поле «Раздел» формы КНИГИ при внесении изменений в форму РАЗДЕЛЫ. Для внесения изменений нужно "перезапустить" форму КНИГИ);

4) после обновления данных в поле «Раздел» формы РАЗДЕЛЫ автоматически открывать форму КНИГИ;

5) продолжать вносить данные о книге. Если при заполнении поля со списком "Автор" нужного автора в списке нет - автоматически открывать форму АВТОРЫ;

6) после открытия формы АВТОРЫ - автоматически закрывать форму КНИГИ;

7) вносить данные об авторе в форму АВТОРЫ;

8) после обновления данных в полях формы АВТОРЫ открывать форму КНИГИ;

9) закончить заполнение формы КНИГИ;

10) после окончания заполнения формы КНИГИ по событию, инициируемому пользователем, открыть форму ЦИТАТЫ. Если дополнять форму ЦИТАТЫ не нужно - переходить к п.12;

11) внести дополнения в форму ЦИТАТЫ;

12) после обновления данных в форме ЦИТАТЫ открыть форму КНИГИ.

#### **Указания по выполнению.**

1) рекомендуется использовать следующие события:

«Отсутствие в списке», «Двойное нажатие кнопки», «После обновления», «Открытие».

2) автоматический процесс внесения изменений не должен запускаться самопроизвольно. Пользователь должен иметь возможность открывать и закрывать формы, не вызывая при этом лавину последующих автоматически выполняемых действий;

3) в необходимых случаях в формах должны быть сделаны подсказки для пользователя.

**Задание 3.** Внесите в таблицу и форму КНИГИ дополнительное поле "КолЭкз" и заполните это поле произвольно, значениями 1, 2 или 3.

Составьте алгоритм и средства для автоматического поиска книги в библиотеке по заявке читателя, содержащей фамилию автора и название книги, и автоматического формирования ответа средствами отчета или формы или с помощью сообщения (программируемое в макросе макрокоманда – Окно сообщения) в зависимости от результатов поиска:

а) К сожалению, такой книги в библиотеке нет; б) Книга имеется в единственном экземпляре. Вы можете получить ее в читальном зале; в) Ваша заявка выполнена.

#### **4.13. Запросы**

##### **Назначение запросов**

- запросы обеспечивают простой доступ к определенному подмножеству записей одной или нескольких таблиц;
- в запросе можно создать вычисляемые поля, которых нет в таблицах, указать условия сортировки и отбора записей;
- на основе запроса можно производить не только выборку, но удаление и обновление записей.

##### **Основные виды запросов**

- на выборку данных;
- на обновление, добавление и удаление записей;
- запросы с вычисляемыми полями;
- запросы с параметром;
- запросы с использованием групповых операций.

##### **Средства создания запросов**

Запрос может быть создан:

- с помощью Мастера запросов;
- в режиме «Конструктор»;
- средствами языка SQL

Запрос на выборку данных предназначен для извлечения из одной или нескольких таблиц БД связанных записей, т.е. записей, относящихся к одному и тому же информационному объекту (студенту, товару и т.д.).

Пример создания запроса на выборку в режиме «Конструктор» (рис. 4.10)

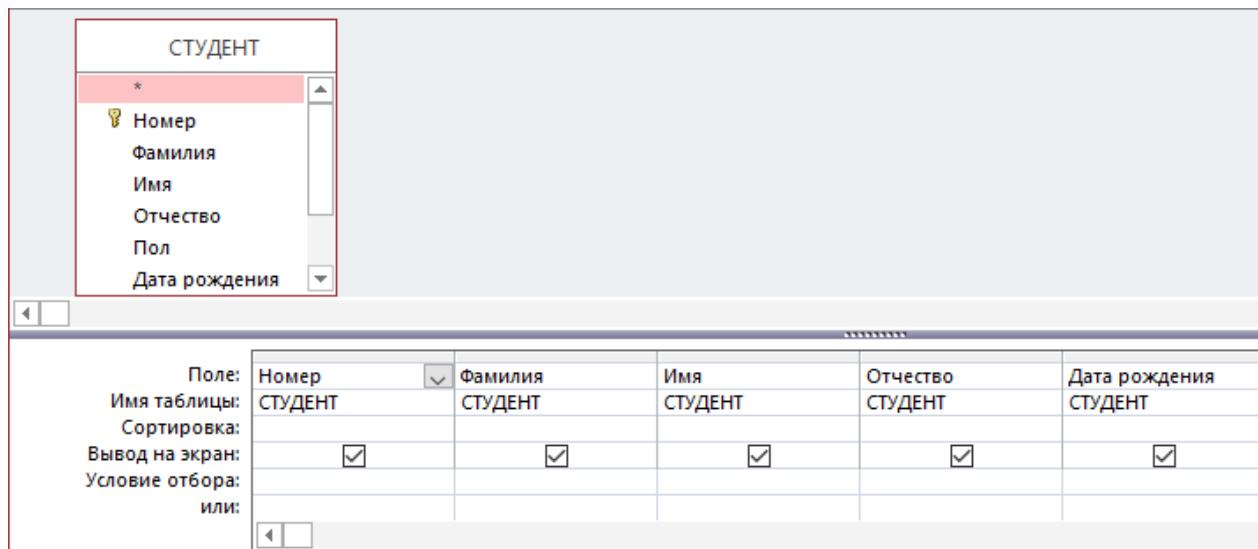


Рис. 4.10.

### **Вычисляемые поля в запросах на выборку**

В запрос можно вводить не только поля из одной или из нескольких таблиц, но и создавать вычисляемые поля на основе имеющихся. Например, если в таблицах имеются поля «Цена» и «Количество», то в запросе можно создать поле

[Цена] \* [Количество].

### **Упражнения 4**

Сформируйте запрос на выборку к таблицам СТУДЕНТ и СЕССИЯ базы данных sess.accdb, включив поля: «Фамилия», «Имя», «Оценка1», «Оценка2», «Оценка3», «Оценка4» и вычисляемое поле «Средний балл».

#### **Указания по выполнению**

При создании в режиме Конструктор запросов вычисляемого поля в заголовке поля записать выражение:

([Оценка1]+[Оценка2]+[Оценка3]+[Оценка4])/4

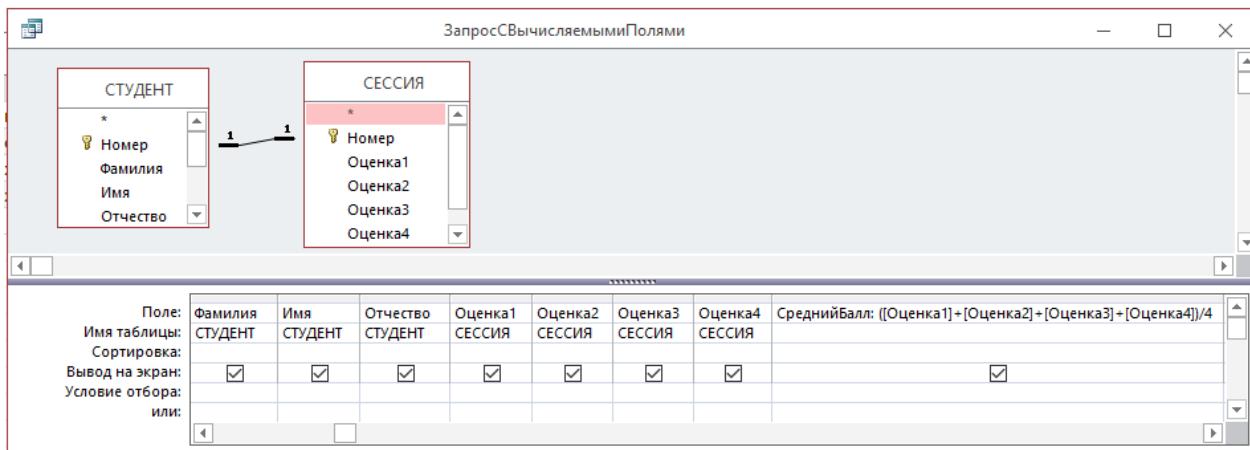


Рис. 4.11.

### Создание запросов с параметрами

При создании запроса в режиме «Конструктор» в условиях отбора можно записать не только выражение с численными значениями, например, <20, но и с параметрами, например: [<имя параметра>]. В последнем случае численное значение параметра будет запрашиваться при открытии запроса.

*Примечание: имя параметра должно отличаться от имени поля.*

На рис. 51 приведен пример запроса с параметром в режиме «Конструктор».

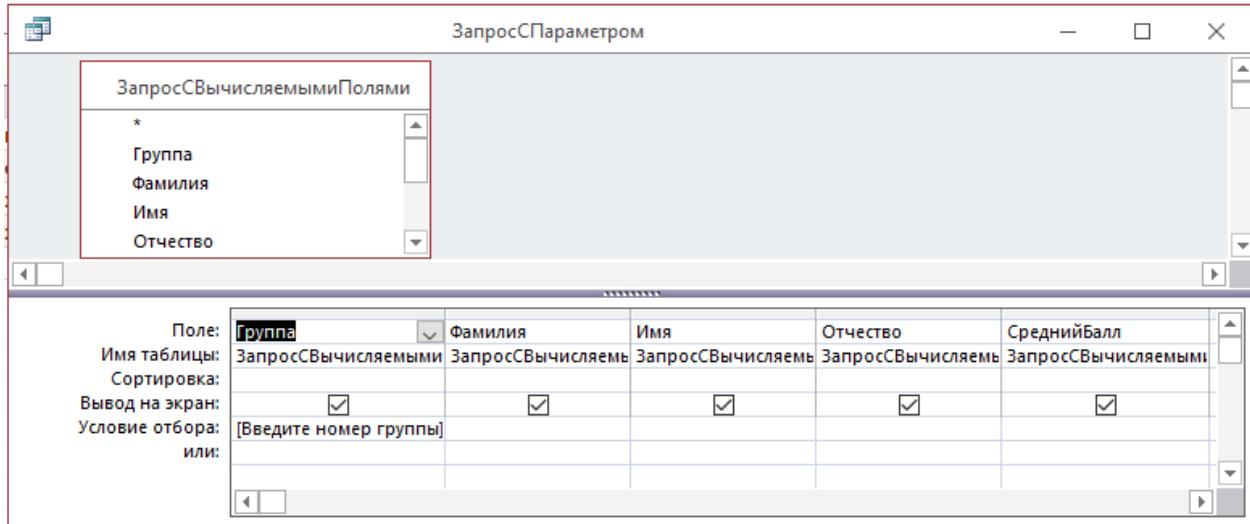


Рис. 4.12.

*Примечание. Основой для создания этого запроса с параметром является не какая-либо таблица базы данных, а запрос с вычисляемым полем «Средний балл», создание которого описано выше.*

## Упражнения 5

Сформируйте запрос с параметром на выборку данных из таблиц СТУДЕНТ и СЕССИЯ, включив поля: «Фамилия», «Имя», «Оценка1», «Оценка2», «Оценка3», «Оценка4», «Дата рождения» и вычисляемое поле «Средний балл» о студентах с датой рождения от [НачДата] до [КонДата].

### Создание запросов с использованием групповых операций

Групповые операции позволяют выделить группы записей с одинаковыми значениями в указанных полях и использовать для других полей этих групп какую-либо статистическую функцию обработки из числа предусмотренных в Access, а именно: Sum - сумма, Avg - среднее значение, Max, Min, Count - число значений поля в группе без учета пустых, StDev - с.к.о. от среднего, Var - дисперсия значений поля в группе, First - значение поля из первой записи, Last - ...из последней.

Порядок создания запроса с использованием групповых операций.

- a) создать запрос на выборку;
- b) #Конструктор# – !Показать или скрыть! – [Итоги]. В результате в макете запроса появится дополнительная строка «Групповая операция», в каждом столбце указана операция – «Групировка»
- c) во всех полях, кроме того, по которому должна производиться группировка, из поля со списком выбрать функцию обработки (Sum, Avg, Max, Min и т.д.).
- d) отредактировать названия и формат данных полей запроса, если это необходимо. Для этого вызвать «Свойства» в контекстно-зависимом меню для соответствующих клеток с названиями полей установить необходимые записи в строках: «Подпись», «Формат поля», «Число десятичных знаков».

Пример запроса с групповыми операциями в режиме Конструктор приведен на рис. 52.

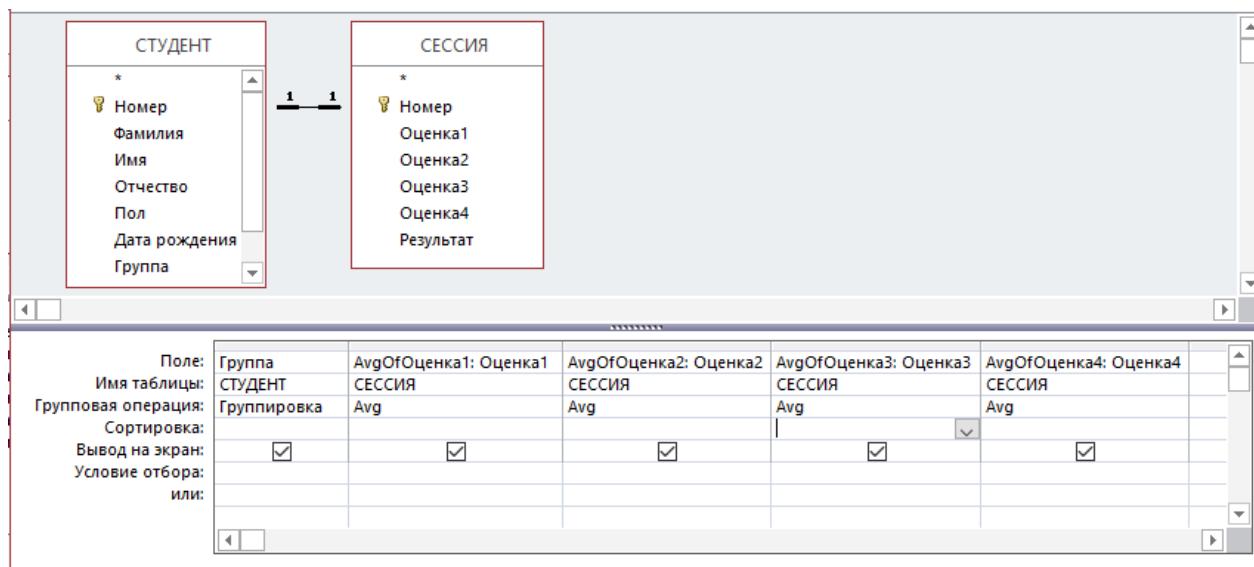


Рис. 4.13.

### Упражнения 6

Сформировать запрос с групповой операцией к таблице СТУДЕНТ для подсчета среднего балла по группе по каждому предмету в отдельности. Запрос сформировать по приведенной в табл. 4.9 форме.

Табл. 4.9.

Группа	ВМ	Физика	Информати	История
10841	4,00	4,33	5,00	4,67
10842	5,00	5,00	4,75	5,00
10843	3,00	4,00	3,00	4,00

### Указания по выполнению.

При редактировании названий и формата полей в запросе по столбцу «Оценка1» воспользуйтесь приведенным ниже на рис. 4.14 в качестве примера образцом записей в панели «Свойства поля»:

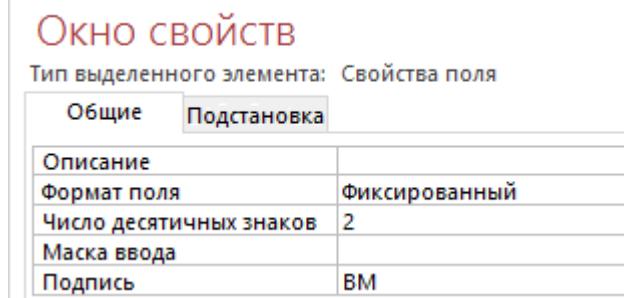


Рис. 4.14

## Запросы на обновление, добавление и удаления записей в таблицах БД

Полезны в случаях, когда изменения в таблицах БД необходимо произвести не по отношению к одной записи, а по отношению к группе записей, связанных некоторой общностью (условием отбора).

Для создания такого запроса вначале создается запрос на выборку, затем в режиме «Конструктор», запрос преобразуется в запрос на обновление, добавление или удаление записей:

#Конструктор# - !Тип запроса! – [Обновление/Добавление/Удаление].

*Примечание. Запрос на добавление записей в БД используется для присоединения записей из одной таблицы к записям другой. Объединяемые таблицы должны иметь одинаковые поля и одинаковые типы данных.*

### Упражнения 7

Сформируйте запрос на выборку данных таблицы СТУДЕНТ, затем преобразуйте его в запрос на обновление и измените с помощью запроса на обновление номер группы с 10841 на 20841 (см. рис. 4.15).

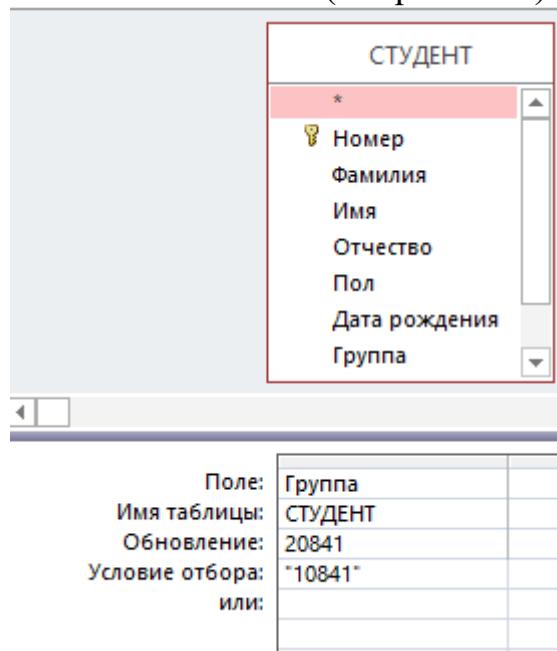


Рис. 4.15.

## Упражнения 8

Создайте таблицу «Абитуриент», предусмотрев в ней те же поля, что и в таблице «Студент», заполните таблицу и произведите объединение этих двух таблиц, создав и запустив запрос на добавление (см. рис.4.16).

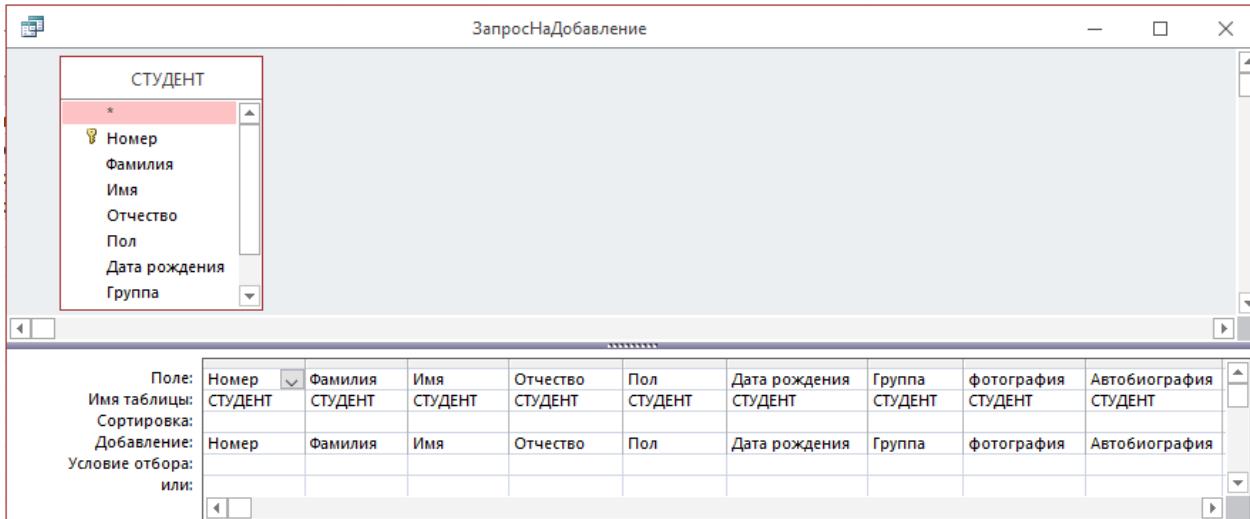


Рис. 4.16.

## Упражнения 9

Сформируйте запрос с параметром на удаление записей из таблицы «Студент» при указании номера группы. Для этого создайте запрос с параметром на выборку, содержащий одно поле – номер группы, затем преобразуйте его в запрос на удаление (рис. 4.17).

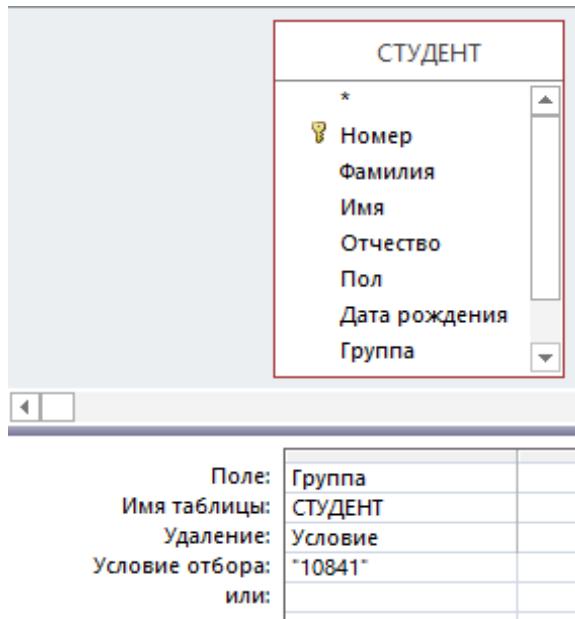


Рис.4.17

#### 4.14. Запросы на языке SQL

SQL (Structured Query Language) является стандартом языков запросов для работы с реляционными базами данных для архитектуры файл-сервер и клиент-сервер, а также в условиях применения системы управления распределенными базами данных. SQL позволяет выполнять:

- выборку данных из таблиц;
- ввод данных в таблицы;
- удаление записей в таблицах;
- обновление (замену) значений полей записи в таблицах.
- создание реляционных таблиц;
- изменение структуры таблиц;
- удаление таблиц;
- соединение таблиц.

Простая конструкция языка SQL для формирования запроса на выборку:

`SELECT таблица1.поле1, .... таблицаk.полек`

AS <псевдоним поля (новый заголовок столбца) или, если создается новое поле, на основе выражения (см. пример 2)(необязательное) >

FROM <источник формирования выборки, например, имя таблицы.  
Если запрос составляется к системе связанных таблиц, то в запросе должны  
быть описаны связи (обязательное)>

WHERE <условие отбора записей (необязательное)>  
GROUP BY <условие группировки (необязательное)>  
ORDER BY <указываются поля, по которым производится сортировка  
(необязательное)>.

*Для формирования запроса к БД на языке SQL:*

Запрос на SQL создается в окне, вызываемом:

- a) #Создание# - !Запросы! – [Конструктор запросов] – [Добавление  
таблицы/Закрыть];
- b) #Конструктор# - !Результаты! – [Режим] – [Режим SQL]

*Для запуска запроса на исполнение:*

#Конструктор# - !Результаты! – [Выполнить]

### **Запрос на выборку к одной таблице**

**Пример 1.** Запрос на выборку полей: «Группа», «Фамилия», «Имя», - таблицы  
СТУДЕНТ.

SELECT Студент.Группа, Студент.Фамилия, Студент.Имя  
FROM Студент;

Ниже в табл. 4.10 приведен результат выполнения запроса.

Табл. 4.10.

Группа	Фамилия	Имя
10842	Панов	Сергей
10843	Сергеев	Петр
10842	Петрова	Анна
10842	Анохин	Андрей
10841	Борисова	Мария
10841	Зайцев	Сергей
10842	Кравцов	Алексей
10841	Шарапова	Мария
*		

*Примечание: если наименование поля состоит из двух или более слов, например «Дата рождения», то это наименование в запросе должно быть заключено в квадратные скобки. Ниже приведен пример такого запроса.*

```
SELECT СТУДЕНТ.Фамилия, СТУДЕНТ.[Дата рождения]
FROM СТУДЕНТ;
```

**Пример 2.** Запрос на выборку полей: «Группа», «Фамилия», «Имя», «Отчество», - таблицы СТУДЕНТ. В результирующей виртуальной таблице, создаваемой в результате выполнения запроса, предусмотрено создание всего двух полей: «Группа» и «ФИО». В поле «ФИО» вводятся фамилия, имя и отчество, разделенные пробелами.

```
SELECT СТУДЕНТ.Группа,Фамилия&" "&Имя&" "&Отчество AS ФИО
FROM СТУДЕНТ;
```

**Упражнение 10.** Сформируйте запрос на выборку полей: «Номер» и «Результат» таблицы СЕССИЯ. Укажите сортировку записей по полю «Номер».

### Запросы на выборку к системе связанных таблиц

Особенность запроса к системе связанных таблиц заключается в том, что в запросе должны быть описаны связи между таблицами БД.

**Пример 1.** Запрос на выборку полей: «Фамилия», «Имя», - таблицы СТУДЕНТ, полей: «Оценка1», «Оценка2», «Оценка3», «Оценка4» - таблицы СЕССИЯ и создание вычисляемого поля «СреднийБалл».

В последней строке запроса указана связь таблиц СТУДЕНТ и СЕССИЯ.

```
SELECT Студент.Группа, Студент.Фамилия, Студент.Имя, Сессия.Оценка1,
Сессия.Оценка2, Сессия.Оценка3, Сессия.Оценка4,
([Оценка1]+[Оценка2]+[Оценка3]+[Оценка4])/4 AS [Средний балл]
FROM Студент INNER JOIN Сессия ON Студент.Номер = Сессия.Номер;
```

**Пример 2.** Запрос с параметром: [НачДата] и [КонДата] на выборку полей: «Фамилия», «Имя», «Дата рождения», - таблицы СТУДЕНТ, полей: «Оценка1», «Оценка2», «Оценка3», «Оценка4» - таблицы СЕССИЯ и создание вычисляемого поля «СреднийБалл». Параметры [НачДата] и [КонДата] определяют условие отбора данных в поле «Дата рождения» таблицы СТУДЕНТ.

```
SELECT СТУДЕНТ.Фамилия, СТУДЕНТ.Имя, СТУДЕНТ.[Дата рождения], СЕССИЯ.Номер,
СЕССИЯ.Оценка1, СЕССИЯ.Оценка2, СЕССИЯ.Оценка3, СЕССИЯ.Оценка4,
([Оценка1]+[Оценка2]+[Оценка3]+[Оценка4])/4 AS СреднийБалл
FROM СТУДЕНТ INNER JOIN СЕССИЯ ON СТУДЕНТ.Номер = СЕССИЯ.Номер
WHERE (((СТУДЕНТ.[Дата рождения])>[НачДата] And (СТУДЕНТ.[Дата рождения])<[КонДата]));
```

**Пример 3.** Запрос с использованием групповых операций к таблицам СТУДЕНТ и СЕССИЯ для группировки по полю «Группа» и подсчета среднего значения по полям: «Оценка1», «Оценка2», «Оценка3», «Оценка4».

```
SELECT СТУДЕНТ.Группа, Avg(СЕССИЯ.Оценка1) AS AvgOfОценка1, Avg(СЕССИЯ.Оценка2)
AS AvgOfОценка2, Avg(СЕССИЯ.Оценка3) AS AvgOfОценка3, Avg(СЕССИЯ.Оценка4) AS
AvgOfОценка4
FROM СТУДЕНТ INNER JOIN СЕССИЯ ON СТУДЕНТ.Номер = СЕССИЯ.Номер
GROUP BY СТУДЕНТ.Группа;
```

## Упражнение 11

**Задание 1.** Сформируйте приведенные в качестве примеров запросы на языке SQL, запустите их на исполнение, убедитесь, что описанные действия по этим

запросам выполняются, затем модифицируйте запросы по заданию преподавателя и вновь убедитесь, что запрограммированные действия выполняются.

**Задание 2.** Создайте на языке SQL запрос на выборку с параметром «№группы» данных полей «Группа», «Фамилия», «Имя», «Отчество» таблицы СТУДЕНТ, создайте вычисляемое поле «Показатель успеваемости», рассчитываемый по данным таблицы СЕССИЯ как среднее арифметическое всех оценок, полученных студентом в сессию.

**Задание 3.** То же что и в задании 1, но с условием сортировки по фамилии.

#### **4.15. Конструирование сложных отчетов**

Данные для сложного отчета удобно готовить на основе запросов, т.к. запрос может содержать вычисляемые поля, а условия отбора могут быть заданы параметрически (см. Создание запросов с параметрами). Дополнительные поля в отчете, содержащие результаты обработки данных, помещенных в отчет по запросу, могут быть созданы в режиме «Конструктор» в самом отчете.

#### **Упражнения 12**

Отчет с промежуточными итогами.

В процессе его создания с помощью Мастера отчетов нужно задать группировку по параметру (например, по параметру «Группа», если создается отчет об успеваемости с промежуточными итогами в виде среднего балла по каждому предмету) и далее в ходе диалога во всплывающей панели рис. 54, вызываемой щелчком левой кнопки мыши на клавише “Итоги...”, нужно задать вид групповой операции.

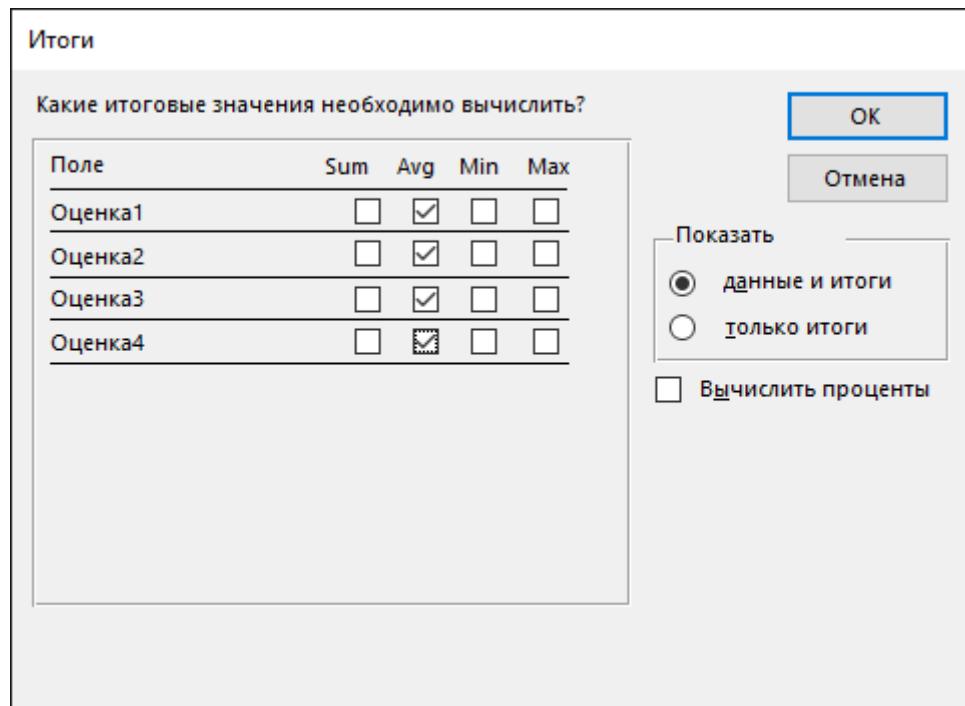


Рис. 4.18.

**Задание 1**

Сформируйте сложный отчет об успеваемости по группам с промежуточными итогами в виде среднего балла по каждому предмету на основе запроса с параметром “№ группы” (см. образец отчета рис.4.19).

**Указания по выполнению**

Для того, чтобы результат вычислений функции Avg был представлен с двумя десятичными знаками, как показано на рис. 4.19, нужно установить параметры поля: «Формат поля» - фиксированный, «Число десятичных знаков» - 2.

Итоги сессии							
Группа	Фамилия	Имя	Отчество	ВМ	Физика	Основы программир.	Информатика
10841	Шарапова	Мария	Сергеевна	4,00	5,00	5,00	4,00
	Зайцев	Сергей	Александрович	4,00	4,00	5,00	5,00
	Борисова	Мария	Михайловна	4,00	4,00	5,00	5,00
Средний балл				4,00	4,33	5,00	4,67
				Декан факультета		Петров В.В.	

Рис. 4.19.

### Задание для самостоятельной работы

Создайте базу данных для учета продаж комплектующих изделий, автоматического оформления счетов, ведения лицевых счетов продавцов. Используйте исходные материалы из файлов: КЕЙ.docx, КЕЙ1.bmp, КЕЙ2.bmp. За основу используйте схему данных рис. 4.20.

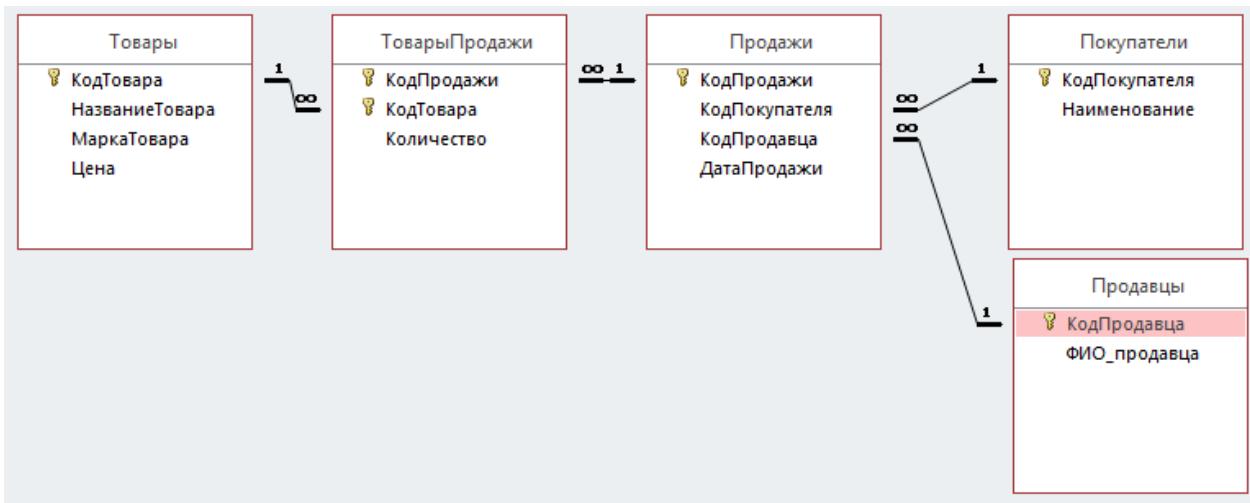


Рис. 4.20.

Оформите форму для ведения продаж по образцу рис.4.21.

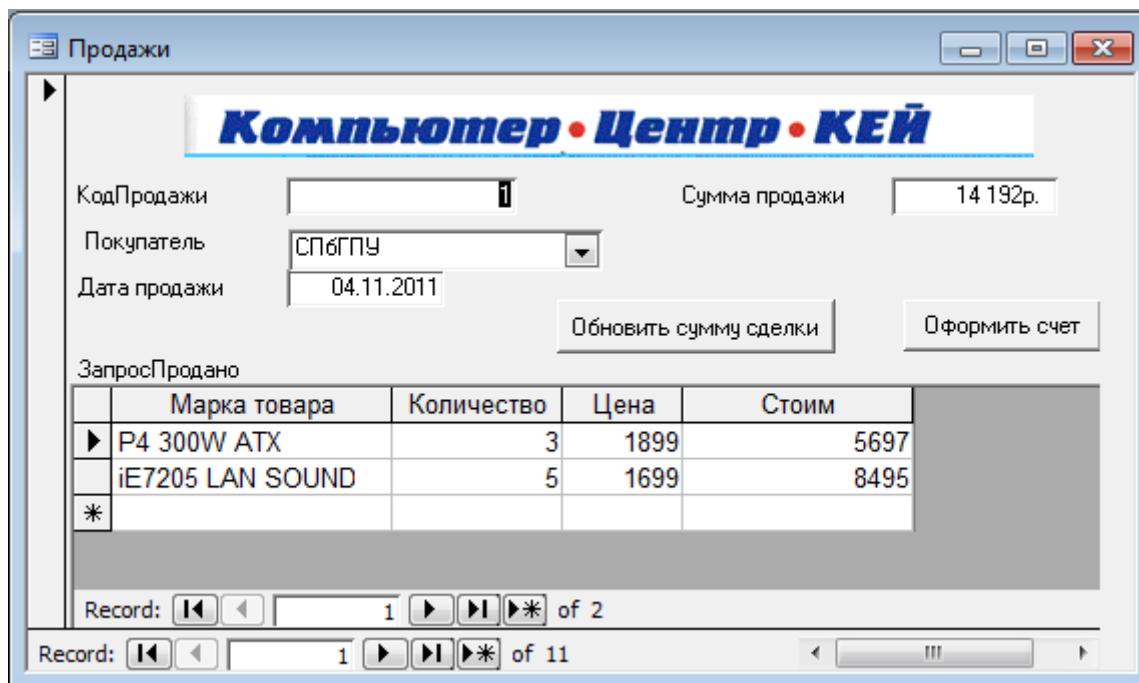


Рис. 4.21.

Счет оформите по образцу рис. 4.22.

 Поставщик: ООО «Кей» ИНН 7825354444 КПП 784601001 ОКВЭД 52.48.13, 51.64.2 ОГРН 1027809259092 ОКПО 44282407 р/с 40702810037040000263 в филиале «ОПЕРУ-4» ОАО «ПСБ» г. С-Петербург к/с 3010181020000000791 БИК 044030791 Телефон для справок : (812) 322-5885																
<b>Счёт №</b> 1 от 04.11.2011																
<b>Покупатель</b> СПбГПУ <b>Лицевой счёт</b> 24545745315																
<table border="1"> <thead> <tr> <th>Модель</th> <th>Количество</th> <th>Цена</th> <th>Сумма</th> </tr> </thead> <tbody> <tr> <td>P4 300W ATX</td> <td>3</td> <td>1899</td> <td>5697</td> </tr> <tr> <td>iE7205 LAN SOUND SATA</td> <td>5</td> <td>1699</td> <td>8495</td> </tr> <tr> <td><b>Общий итог</b></td> <td></td> <td></td> <td><b>14192</b></td> </tr> </tbody> </table>	Модель	Количество	Цена	Сумма	P4 300W ATX	3	1899	5697	iE7205 LAN SOUND SATA	5	1699	8495	<b>Общий итог</b>			<b>14192</b>
Модель	Количество	Цена	Сумма													
P4 300W ATX	3	1899	5697													
iE7205 LAN SOUND SATA	5	1699	8495													
<b>Общий итог</b>			<b>14192</b>													
<i>Директор магазина</i> <b>Соловьев С.С.</b>																

Рис. 4.22.

## Указания по выполнению

1. Для формирования макета счета используйте файл *rekvizit.docx* и файл с логотипом магазина КЕЙ.bmp.
2. Реляционную форму для ведения продаж создайте по образцу рис.4.21 на основе таблицы «Продажи» и запроса «ЗапросПродано», в котором создайте вычисляемое поле Стоимость=Цена\*Количество.
3. Кнопка «Обновить сумму сделки» должна запускать макрос, содержащий команду ЗадатьЗначение<sup>21</sup> с параметрами:  
Элемент - [Forms]![Продажи]![Сумма продажи]  
Выражение -  
`DSum("[Цена]*[Количество]";"[ЗапросПродано]";"[КодПродажи]=[Forms!Продажи]![КодПродажи]"")`  
Кнопка «Оформить счет» должна запускать макрос, открывающий отчет, сформированный на основе запроса «ПродажиЗапрос». В поле «Условия отбора» команды «ОткрытьОтчет» макроса записать условие: [ПродажиЗапрос]![КодПродажи]=[Forms]![Продажи]![КодПродажи].
4. Для автоматического формирования даты используйте Общие выражения/ Текущая дата (=Дата) в Построителе выражений. Можно просто ввести выражение «=Дата» в элементе «Поле».

### 4.16. Разработка приложений пользователя на VBA в среде Access

Visual Basic for Applications (VBA) относится к языкам объектно-ориентированного программирования. VBA имеет значительно большие функциональные возможности по сравнению с макросами, а именно:

- VBA как всякий язык программирования имеет значительно более широкие возможности управления логикой выполняемой программы;

---

<sup>21</sup> Эта макрокоманда отсутствует в основном списке. Для ее появление в списке нужно активизировать кнопку «Показать все действия» на вкладке «Конструктор»

- VBA позволяет использовать в выражениях и в качестве аргументов процедур переменные, значения которых могут динамически изменяться, в макросах же применяются только статические значения;
- VBA обеспечивает очень гибкие возможности управления всеми объектами Access, особенно объектами, связанными с данными. Например, VBA дает возможность работать с отдельными записями набора. Макросы позволяют работать только с набором записей.
- VBA обеспечивает возможность создавать средства перехвата и обработки ошибок. Макросы этого не позволяют.
- VBA поддерживают несколько механизмов для взаимодействия с другими приложениями, такими как Automation (Автоматизация), OLE (связывание и внедрение объектов), DDE (динамический обмен данными). Макросы позволяют только запустить другое приложение с помощью макрокоманды «Запуск приложения» (RunApp).

Поэтому макросы целесообразно использовать для создания прототипов приложений пользователя, а «доведение до ума» приложений следует производить с использованием VBA.

### **Основные средства VBA в Access**

**ОБЪЕКТ.** Библиотека встроенных объектов VBA в Access содержит большое количество объектов, находящихся на различных уровнях иерархии. Примеры объектов: Application (Приложение), Forms (Формы), Reports (Отчеты), Tables (Таблицы), Queries (Запросы), Docmd (). При ссылке на объект нужно указывать не только имя объекта, но и "путь к нему", например:

Application.CurrentProject.AllForms("Сотрудники")

Application.Forms("Вакансии")

Forms!Вакансии – эта форма записи равнозначна предыдущей.

Forms("Подразделение") ("Код подразделения")

Специальный объект, который позволяет в программах VBA выполнять макрокоманды – Docmd.

Ссылку на объект можно упростить, если использовать объектные переменные, например: Set ttt = Forms!Штаты!Название подразделения.

Если нужно запрограммировать подряд несколько действий, то "путь к объекту" можно указать один раз, используя конструкцию With - End With, например:

With rs

.Дата рождения=#11/1/72#

```
.Образование="высшее"
End With
Здесь rs – объектная переменная.
```

## СВОЙСТВА И МЕТОДЫ ОБЪЕКТОВ

Каждый объект имеет свойства (например, Visible), которые являются его характеристиками и методы, позволяющие управлять поведением этого объекта. Синтаксис применения метода:

Объект.Метод(<список аргументов>).

Примеры команд программы, определяющих действия над объектами:  
Docmd.OpenForm "Orders", acNormal, ,stLinkCriteria –открыть форму Orders.  
Здесь acNormal – встроенная константа Access.

Docmd.OpenReport "Вакансии", acPreview

В данном примере открывается отчет ВАКАНСИИ в режиме предварительного просмотра.

В следующем примере метод OpenRecordset объекта Database возвращает ссылку на объект Recordset (Набор записей), присваиваемую объектной переменной rs.

```
Dim db As Database, As Recordset
Set db=CurrentDB
Set rs = db.OpenRecordset ("Сотрудники")
```

```
.....
rs.Close
db.Close
```

**СВОЙСТВО** представляет атрибут объекта, определяющий его характеристики, например, цвет или размер. Синтаксис команды установки значения свойства: Объект.Свойство=ЗначениеСвойства.

Примеры:

```
Forms ("Подразделение") ("КодПодразделения").Visible = False
Me! КодПодразделения.Visible = False
```

В приведенном выше примере использовано специфическое свойство форм и отчетов **Me**. Оно представляет собой ссылку на форму или отчет, программа которого выполняется в данный момент.

Следующий пример показывает каким образом можно программным путем прочитать свойство объекта:

```
intCount = Forms.Count
```

**СОБЫТИЕ** представляет собой действие (например, щелчок мышью, нажатие клавиши), распознаваемое объектом. С событием можно связать процедуру на VBA, которая будет запускаться на исполнение при возникновении события. Синтаксис заголовка процедуры обработки события:

Private Sub <имя объекта>\_<имя события>

Пример заголовка процедуры обработки события инициализации объекта:

Private Sub Class\_Initialize()

Программа на VBA может содержать ключевые слова, операторы (присваивания, безусловного и условного перехода, выбора варианта, цикла), переменные, константы и выражения.

## Применение VBA в Access

VBA может быть использован для создания:

- стандартных модулей (которые не связаны с конкретным приложением, а могут быть использованы в любом приложении);
- модулей класса двух типов:
  - базовых модулей класса;
  - модулей форм и отчетов.

## Открытие окна для разработки программ на VBA

Для создания программ на языке Visual Basic for Application или пользовательских форм нужно:

- a) Открыть окно разработки программ: #Работа с базами данных# - !Макрос! – [Visual Basic]; В результате откроется окно для создания программного кода. В открывшемся окне будет содержаться раздел “Declaration”, в котором по умолчанию выводятся инструкции “Option Compare Database” и “Option Explicit”. Первая определяет способ сравнения строк, а вторая указывает на необходимость явного объявления переменных в модуле.
- b) [Insert] – [Procedure...].

- c) ввести в поле “Name:” всплывающей панели “Add Procedure” имя процедуры.
- d) задать тип процедуры: “Private” или “Public”. Тип процедуры определяет уровень видимости: уровень модуля и уровень проекта соответственно. Процедура, объявленная “Private”, может быть вызвана только из того модуля, в котором она написана. Более того, это объявление запрещает ее использование как самостоятельной процедуры; она может быть вызвана только из другой процедуры. Процедура, объявленная “Public”, может быть вызвана из любого модуля проекта.

При описании процедур и функций может быть использовано ключевое слово “Static”. Оно не влияет на область действия процедуры, но воздействует на все переменные, описанные внутри этой процедуры или функции. В этом случае все локальные переменные получают статус Static и, тем самым, сохраняются в памяти после завершения процедуры и при повторном ее вызове сохраняют свои прежние значения.

## **Создание модуля**

В окне редактора кода VBA создать текст модуля, затем:  
[File] – [Save] – {имя модуля}.

Пример модуля.

```
Private Sub КолЗаказаТовара_AfterUpdate()
Dim stDocName As String
Dim stLinkCriteria As String
stDocName = "Склад"
stLinkCriteria = "[КодТовара] = " & Me![КодТовара]
Продано = Forms("Расход")("КолЗаказаТовара").Value
DoCmd.OpenForm ("Склад"), , , stLinkCriteria
Остаток = Forms("Склад")("Количество").Value
Forms("Склад")("Количество").Value = Остаток - Продано
Forms("Склад")("Стоимость").Value = Forms("Склад")("Цена").Value * 
Forms("Склад")("Количество").Value
```

End Sub

Этот модуль выполняет перерасчет остатка и стоимости товаров на складе после продажи.

## Программирование форм и отчетов

Все формы и отчеты, хранящиеся в базе данных Access образуют два семейства – AllForms и AllReports, которые подчиняются объекту CurrentProject. Каждый элемент семейства AccessObject – представляет собой форму или отчет, включая и подчиненные формы и отчеты. Обращаться к конкретной форме или отчету можно или по имени или по индексу, например

`CurrentProject.AllForms("Работники")`

`CurrentProject.AllForms(5)`

Можно для доступа к форме или отчету использовать объектную переменную, которая должна быть объявлена, например, так:

`Dim frm As AccessObject`

Кроме семейств AllForms и AllReports имеются семейства Forms и Reports. Они включают только открытые в данный момент формы или отчеты.

Для открытия формы или отчета из программы на VBA используется метод OpenForm специального объекта Docmd:

`Docmd.OpenForm <имя формы>[<параметры>]`

Чтобы закрыть форму или отчет используется метод Close объекта Docmd.

Каждый объект Form содержит семейство Controls (элементов управления: командных кнопок, текстовых окон, окон со списком и т. д.).

Обращение к объекту семейства Controls в форме производится по схеме:

`Forms!<имя формы>!<имя элемента Cotrols>`

Пример:

`Forms!Продажи!Стоимость`

Каждый элемент семейства Controls имеет закрепленную за ним константу:

Табл. 4.11.

Выключатель	acToggleButton
Группа переключателей	acOptionGoup
Кнопка	acCommandButton
Линия	acLine
Набор вкладок	acTabCtl
Надпись	acLabel
Переключатель	acOptionButton

Подчиненная форма	acSubForm
Поле	acTextBox
Поле со списком	acComboBox
Пользовательский элемент управления	acCustomControl
Присоединенная рамка объекта	acBoundObjectFrame
Прямоугольник	acRectangle
Рисунок	acImage
Свободная рамка объекта	acObjectFrame
Список	acListBox
Страница	acPage
Флажок	acCheckBox

### Пример Access-приложений, разработанных с использованием VBA

В приложении 1 решается задача учета продаж товаров с автоматической корректировкой остатков товара и остаточной стоимости товаров на складе. На рис. 4.23 приведена возможная схема данных.

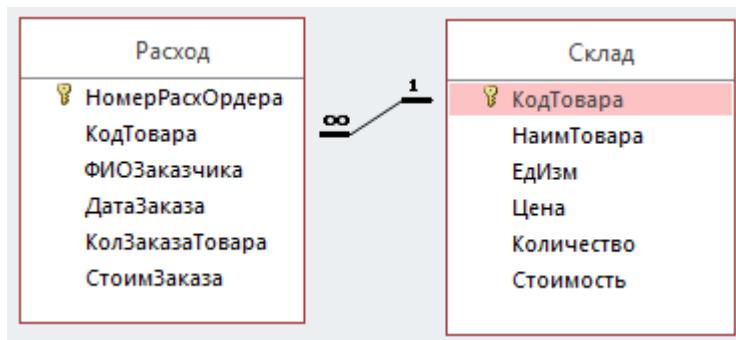


Рис. 4.23.

Первая процедура, запускаемая по событию в поле «КолЗаказаТовара», формы РАСХОД, обеспечивает корректировку сведений о наличии товара на складе после продажи (поле «Количество» формы «Склад»)

### Option Compare Database

```

Private Sub КолЗаказТовара_AfterUpdate()
Dim stDocxName As String
Dim stLinkCriteria As String
stDocxName = "Склад"
stLinkCriteria = "[КодТовара] = " & Me![КодТовара]
Продано = Forms("Расход")("КолЗаказТовара").Value
'Следующий оператор открывает форму СКЛАД на записи, связанный по
'ключевому полю КодТовара с записью, открытой в данный момент
'в форме Расход
Docxmd.OpenForm ("Склад"), , , stLinkCriteria
Остаток = Forms("Склад")("Количество").Value
Forms("Склад")("Количество").Value = Остаток - Продано
Forms("Склад")("Стоимость").Value=Forms("Склад")("Цена").Value*Forms("Склад")("Количество").Value
End Sub
Следующая процедура производит автоматический перерасчет остаточной
стоимости каждого товара после изменения его количества на складе (в том
числе и по результатам продажи) и при изменении цены.

```

### Option Compare Database

```

Private Sub Количество_AfterUpdate()
Forms("Склад")("Стоимость").Value
=Me("Цена").Value*Me("Количество").Value
End Sub
Private Sub Цена_AfterUpdate()
Forms("Склад")("Стоимость").Value
=Me("Цена").Value*Me("Количество").Value
End Sub

```

В приложении 2 с использованием макросов и модулей решается задача внесения данных в базу данных с внешней аппаратурой. БД предназначена для учета отгрузки предприятием продукции. Взвешивание отгружаемой продукции производится на автоматических весах. Результат взвешивания

считывается в компьютер с автоматических весов и записывается в файл. Программа чтения данных представляет собой EXE-модуль. Запуск EXE-модуля и перенос данных из файла в БД осуществляется из БД.

Запуск процесса чтения результата взвешивания может быть организован из формы, с помощью которой производится занесение результатов взвешивания в таблицу базы данных. Форма может иметь вид рис. 4.24.

Рис. 4.24.

С помощью кнопки управления «Взвешивание» запускается макрос, который, в свою очередь, запускает модуль на VBA:

```

Dim a, b, p As Variant
Shell "c:\scales.exe" 'Запуск ехе-модуля, выполняющего
'чтение данных с автоматических весов и запись в файл b1.dat
a = Timer 'Запуск функции чтения кода времени с таймера
b = a
While (b - a < 2) 'Цикл ожидания на 2 сек
    b = Timer      '(За 2 сек производится взвешивание)
Wend
Open "D:\b1" For Input As 1 'Открытие файла b1.dat для чтения
Input #1, p    'Чтение результата взвешивания p из файла
Close #1       'Закрытие файла
Beep          'Звуковой сигнал
Forms![Отгрузка]![масса,т] = p * 0.01 'Ввод данных в БД
Forms![Отгрузка]![Дата/время] = Now() 'Ввод текущего времени в БД

```

## Ответы к заданиям по разделу «Макросы»

### Задание 2.

Табл. 4.12.

Форма	Эл-т управле- ния	Событие	Макрос	Примечание
КНИГИ	Форма	Открытие	ЗакрФормыРазделы	
	Раздел	Двойное нажатие кнопки	ОткрФормыРазделы	Подсказка в поле "Примеч. формы"
	Форма	После обновления	ОткрФормыЦитаты	
КНИГИ/ АВТОРЫ подчинен. форма	Форма	Двойное нажатие кнопки	ОткрФормыАвторы	Подсказка в поле "Примеч. формы" КНИГИ
РАЗДЕЛЫ	Форма	Открытие	ЗакрФормыКниги	
	Форма	После обновления	ОткрФормыКниги	
АВТОРЫ	Форма	Открытие	ЗакрФормыКниги	
	Форма	После обновления	ОткрФормыКниги	
ЦИТАТЫ	Форма	После обновления	ОткрФормыКниги	

### Задание 3.

Последовательность действий:

1. Создать макросы: «Ответ1», «Ответ2» и «Ответ3» с сообщениями:

- а) К сожалению, такой книги в библиотеке нет;
- б) Книга имеется в единственном экземпляре. Вы можете получить ее в читальном зале;
- с) Ваша заявка выполнена

соответственно.

Для создания каждого из макросов:

- а) #Создание# – !Макросы и код! – [Макрос];
- б) Выбрать макрокоманду «Окно сообщения» и ввести в качестве параметра один из текстов, приведенных в п.п. 1 а-с.

2. Создать запрос с параметром «ЗапросЧитателя» к таблицам КНИГИ и АВТОРЫ, включив в запрос поля «Фамилия», «Имя» из таблицы АВТОРЫ, поля «Название», «КолЭкз» из таблицы КНИГИ.

В запросе установить условия отбора [ФамАвт], [ИмяАвт], [НазвКниги] по полям «Фамилия», «Имя» таблицы АВТОРЫ и полю «Название» таблицы КНИГИ.

3. Создать форму «ЗапросЧитателя» на основе этого запроса с параметром.

*Внимание! Поле «КолЭкз» из запроса «ЗапросЧитателя» в форму «ЗапросЧитателя» не включать!*

4. Создать макрос «ЗапросЧитателя» в виде последовательности макрокоманд с проверкой условий (см. рис. 4.25<sup>22</sup>).

При формировании условий можно использовать построитель выражений.

5. Создать в главной кнопочной форме кнопку запуска макроса «ЗапросЧитателя».

```

ОткрытьФорму
Имя формы ЗапросЧитателя
Режим Форма
□ Если [Формы]![ЗапросЧитателя]![КолЭкз]>1 то
    ЗапускМакроса
        Имя макроса Ответ3
□ Иначе
    □ Если [Формы]![ЗапросЧитателя]![КолЭкз]=1 то
        ЗапускМакроса
            Имя макроса Ответ 2
    □ Иначе
        ЗапускМакроса
            Имя макроса Ответ1
        Конец блока "Если"
Конец блока "Если"

```

Рис. 4.25.

---

<sup>22</sup> Неиспользуемые строчки макрокоманд не приведены.

#### **4.17. Защита баз данных.**

В Access 2016 существуют следующие способы защиты информации:

- защита базы данных (accdb-файла) с помощью пароля и шифрования;
- защита приложения Microsoft Access путем скрытия объектов в окне базы данных и настройки параметров запуска;
- защита паролем программы на языке VBA;
- защита программ VBA путем создания файла, в котором отсутствует программный код;
- защита базы данных и ее объектов средствами защиты на уровне пользователей.

##### **Защита базы данных (accdb-файла) с помощью пароля**

Чтобы установить пароль доступа к базе данных, нужно открыть базу данных в монопольном режиме. Для этого нужно выполнить следующую последовательность действий:

- a) #Файл# – [Открыть] – [Обзор];
- b) найти в каталогах файл базы данных и пометить его;
- c) [Открыть] – [Монопольно].

После открытия окна базы данных сохранить ее с паролем:

- d) #Файл# - [Сведения] – [Зашифровать с использованием пароля],
- e) задать пароль доступа в окно всплывающей панели «Задание пароля базы данных».

Для снятия пароля базы данных:

- a) #Файл# – [Открыть] – [Обзор];
- b) найти в каталогах файл базы данных и пометить его;
- c) [Открыть] – [Монопольно].

После открытия окна базы данных:

- d) #Файл# - [Сведения] – [Расшифровать базу данных],
- e) ввести ранее установленный пароль доступа в окно всплывающей панели «Удаление пароля базы данных».

### **Скрытие объектов и окна базы данных**

Для скрытия объектов (таблиц, запросов, форм, отчетов, макросов, модулей) нужно:

- a) пометить объект (таблицу, запрос, отчет и т. д.) в Области навигации «Все объектыAccess», затем вызвать контекстное меню, выбрать пункт «Скрыть в этой группе». В результате значок скрытого объекта станет неконтрастным;
- b) #Файл# – [Параметры] – [Текущая база данных] – [Навигация/Параметры навигации];
- c) снять флажок «Показывать скрытые объекты».

Для того, чтобы открыть скрытые объекты, нужно выполнить следующую последовательность действий:

- a) #Файл# – [Параметры] – [Текущая база данных] – [Навигация/Параметры навигации];
- b) установить флажок «Показывать скрытые объекты».

В результате значок скрытого объекта появится, но будет неконтрастным. Для восстановления скрытого объекта нужно пометить его значок в Области навигации «Все объектыAccess», затем вызвать контекстное меню, выбрать пункт «Показать в этой группе».

Можно скрыть само окно базы данных (Область навигации). Для этого нужно выполнить следующую последовательность действий:

- a) #Файл# – [Параметры] – [Текущая база данных] – [Навигация/Параметры навигации] – [Категории/Тип объекта] ;
- b) снять флагки объектов в окне «Группы для «Тип объекта»».

Для того, чтобы увидеть скрытое окно базы данных, нужно выполнить следующую последовательность действий:

- a) #Файл# – [Параметры] – [Текущая база данных] – [Навигация/Параметры навигации] – [Категории/Тип объекта] ;
- b) установить флагки объектов в окне «Группы для «Тип объекта»».

### Защита путем создания файла ACCDE базы данных

Файл ACCDB, в котором хранится база данных Access 2016, может быть преобразован в файл ACCDE. В файле ACCDE весь программный код базы данных хранится в скомпилированном виде. Это исключает возможность внесения изменений в модули, формы и отчеты базы данных в режиме Конструктор и, таким образом, частично защищает базу данных от несанкционированных изменений.

*Примечание. ACCDE. Расширение имени файла для файлов Office Access 2016, сохраняемых в режиме «только выполнение». В ACCDE-файлах удален исходный код Visual Basic для приложений (VBA) (VBA (Visual Basic for Applications). Версия макроязыка программирования Microsoft Visual Basic, используемая для программирования приложений для Microsoft Windows и поставляемая с некоторыми программами корпорации Майкрософт.). Пользователь ACCDE-файла может только выполнять, но не изменять программу на языке VBA.*

Для того, чтобы преобразовать файл ACCDB в ACCDE нужно открыть базу данных, затем:

- a) #Файл# – [Сохранить как] – [Сохранить базу данных как] – [Создать ACCDE];.

*Примечание. Файл ACCDE нельзя вновь преобразовать в ACCDB! Поэтому, чтобы сохранить возможность внесения изменений в базе данных, перед преобразованием файла ACCDB в ACCDE нужно сохранить резервную копию файла ACCDB!*

## 5.PowerPoint 2016

Приложение PowerPoint позволяет создавать, редактировать, демонстрировать и выводить на печать презентации. Презентация представляет собой последовательность слайдов (кадров). Слайд может включать графическое изображение, текст, звук, эффекты анимации. В состав слайда могут быть включены объекты (таблицы, диаграммы, формулы, созданные в других приложениях WINDOWS 10/8/7: Word, Excel, Paint и др.).

Запуск программы Microsoft PowerPoint в WINDOWS 10 можно произвести из главного меню:

[Пуск] – [Все приложения] – [PowerPoint 2016].  
В результате откроется окно PowerPoint (рис. 5.1).

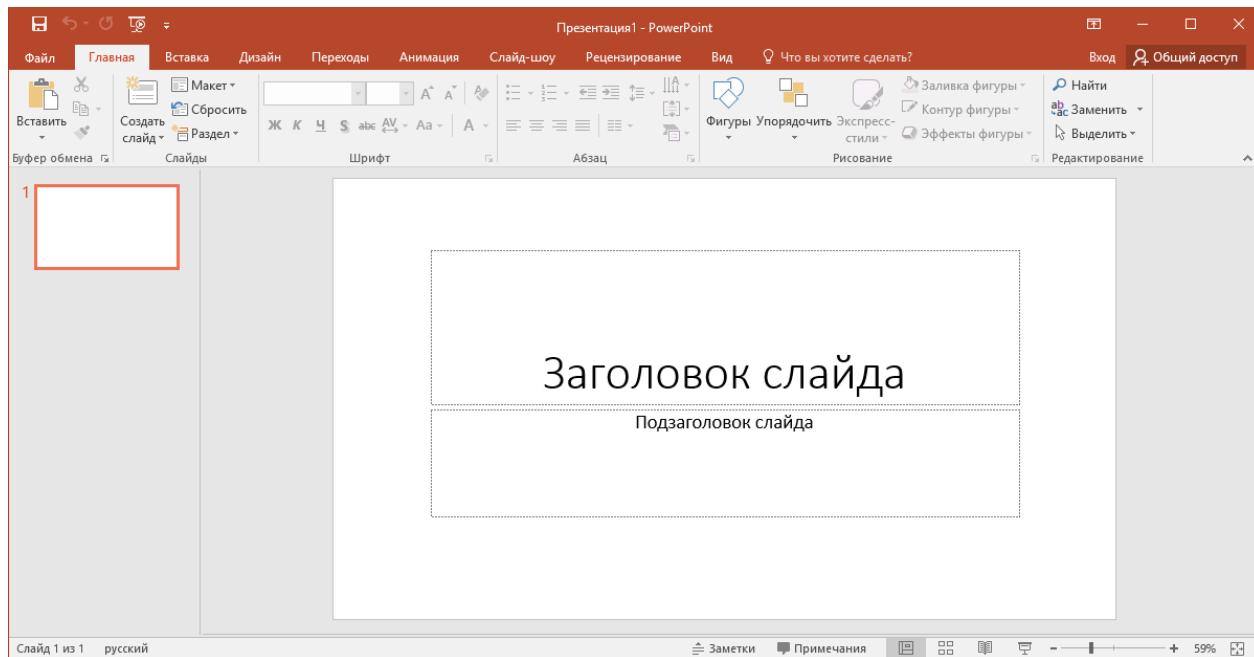


Рис. 5.1

### 5.1. Создание титульного слайда и файла презентации

- #Файл# - [Создать] ;
- во всплывающей диалоговой панели «Создать» выбрать вид шаблона оформления, если шаблон не нужен – выбрать «Пустая презентация» ;
- [Создать].
- выбрать дизайн презентации:

- #Дизайн# – !Темы!  
затем выбрать дизайн из предлагаемой коллекции;
- e) установить при необходимости стиль фона, используя пиктограммы в группе «Фон»;
  - f) заполнить заголовок и подзаголовок титульного макета слайда;
  - g) сохранить файл презентации:  
#Файл# - [Сохранить] - {имя файла} – [Сохранить].

## **5.2. Добавление слайда в состав презентации**

#Главная# - !Слайды! - [Создать слайд]

## **5.3. Создание слайда**

После добавления слайда в состав презентации нужно:

- a) выбрать в области задач вид макета слайда. Для этого вызвать контекстное меню, выбрать пункт «Макет», затем во всплывающей коллекции макетов выбрать необходимый вид макета;
- b) ввести текст заголовка, текст в основном поле, вставить объекты (текстовые фрагменты, таблицы, диаграммы, рисунки) в соответствующие поля макета слайда (см. раздел «Создание объектов в слайдах»).

### **Создание объектов в слайдах**

В поле, предназначенном для текста, можно ввести текст с клавиатуры или вставить фрагмент текста из документа Word.

В поле, предназначенном для диаграмм, можно создать диаграмму PowerPoint или вставить диаграмму, созданную в Excel 2016.

В поле, предназначенном для таблиц, можно создать таблицу PowerPoint или загрузить таблицу, созданную в Excel 2016 или Word.

В любое место поля слайда можно вставить объект WordArt (фигурный текст), стандартный рисунок (картинку) или рисунок из файла.

### ***Вставка текстового фрагмента из документа Microsoft Word***

- a) открыть документ Word, содержащий текстовый фрагмент;
- b) пометить текстовый фрагмент;
- c) скопировать текстовый фрагмент в буфер промежуточного хранения:  
#Главная# - !Буфер обмена! – [Копировать]<sup>23</sup>;
- d) открыть файл презентации или создать его;
- e) выбрать слайд, в который необходимо вставить текстовый фрагмент;
- f) пометить поле макета слайда, предназначенное для вставки текста;
- g) #Главная# - !Буфер обмена! – [Копировать];
- h) отредактировать формат текста на слайде.

### ***Вставка диаграммы PowerPoint***

PowerPoint содержит встроенные средства для создания диаграмм. Для вставки диаграммы PowerPoint нужно:

- a) перейти в режим добавления слайдов:  
#Главная#!Слайды! - [Создать слайд];
- b) выбрать вид макета слайда, содержащего поле для вставки таблицы и диаграммы;
- c) пометить поле макета слайда, предназначенное для вставки таблицы/диаграммы, содержащее пиктограмму «добавление диаграммы»;
- d) сделать щелчок левой кнопкой мыши на пиктограмме «добавление диаграммы»;
- e) выбрать вид диаграммы из коллекции во всплывающем диалоговом окне «Вставка диаграммы»;

В результате пользователю будет предъявлен макет таблицы с данными, внешне похожей на таблицу Excel 2016, и диаграмма, построенная по этим данным и «вставленная» в слайд. Нужно изменить содержимое таблицы (внести в нее все необходимые заголовок, шапку, боковик и данные). Все вводимые данные будут отображены на диаграмме.

---

<sup>23</sup> Операции копирования, вырезания и вставки могут выполняться с использованием контекстного меню (пример приведен в разделе «Вставка диаграммы Microsoft Excel 2016») или с помощью клавиш клавиатуры: <Ctrl+C>, <Ctrl+X>, <Ctrl+V> соответственно.

### ***Вставка диаграммы Microsoft Excel 2016***

- a) открыть рабочую книгу Excel 2016, содержащую диаграмму, или, открыв новую рабочую книгу, создать диаграмму;
- b) в поле диаграммы вызвать контекстное меню, в меню выбрать пункт «Копировать»;
- c) открыть файл презентации или создать его;
- d) открыть или создать слайд, в который нужно вставить диаграмму;
- e) в поле макета слайда, предназначенное для вставки диаграммы вызвать контекстное меню, в меню выбрать пункт «Вставить».

### ***Вставка таблицы Microsoft Excel 2016***

- a) открыть документ Excel 2016, содержащий таблицу, или создать новую рабочую книгу Excel 2016 и создать таблицу;
- b) пометить таблицу;
- c) в поле диаграммы вызвать контекстное меню, в меню выбрать пункт «Копировать»;
- d) открыть файл презентации или создать его;
- e) открыть или создать слайд, в который нужно вставить таблицу;
- f) в поле макета слайда, предназначенное для вставки таблицы вызвать контекстное меню, в меню выбрать пункт «Вставить».

### ***Вставка объектов WordArt***

- a) открыть файл презентации или создать его;
- b) открыть или создать слайд, в который нужно вставить таблицу;
- c) создать объект WordArt. Для этого:
  - #Вставка# - !Текст! – [WordArt];
  - выбрать стиль написания символов из коллекции экспресс-стилей;
  - ввести с клавиатуры текст;
  - доработать стиль написания символов, если необходимо, для чего:

- #Формат# - !Стили WordArt! – [Заливка текста/Контур текста/Анимация];
- выбрать из коллекций необходимый стиль.

Можно сконструировать свой стиль. Для этого:

#Формат# - !Стили WordArt! – [Открыть диалоговое окно]<sup>24</sup>;

- d) отредактировать размер и размещение объекта WordArt в поле слайда.

### ***Вставка рисунков***

- открыть файл презентации или создать его;
- открыть или создать слайд, в который нужно вставить рисунок;
- #Вставка# - !Иллюстрации! – [Рисунок];
- во всплывающем диалоговом окне «Вставка рисунка» выбрать файл рисунка;
- [Вставить].

### ***Вставка видео- и звуковых клипов***

- открыть файл презентации или создать его;
- открыть или создать слайд, в который должен быть вставлен видео- или звуковой клип;
- #Вставка# - !Иллюстрации! – [Клип];
- пользуясь средствами управления всплывающей области «Клип» выбрать нужный Media – файл.

### ***Вставка объектов, созданных в других приложениях Windows***

Вставка производится стандартным способом через буфер промежуточного хранения (см., например, раздел «Вставка диаграммы Microsoft Excel 2016»). В частности, можно нажатием клавиши «PrtScr» занести в буфер промежуточного хранения графический экран. Нажатием клавиш «Alt+PrtScr» в буфер промежуточного хранения заносится изображение активного диалогового окна, находящегося на экране монитора.

---

<sup>24</sup> Для этого нужно сделать щелчок левой кнопкой мыши на пиктограмме  в группе !Стили WordArt!

Отредактировать графическое изображение (например, вырезать только нужную область изображения) можно в графическом редакторе Paint, после чего вырезанную часть поместить в буфер промежуточного хранения.

#### **5.4. Добавление слайда из файла**

Слайд может быть добавлен **только** из файла с расширением ppt. Создать его можно так же, как описано в разделе «Создание титульного слайда и файла презентации».

Для добавления нужно:

- a) открыть презентацию, содержащую слайд, который нужно добавить и пометить слайд;
- b) #Главная# - !Буфер обмена! – [Копировать];
- c) открыть презентацию, в которую нужно вставить слайд и пометить слайд, после которого должен быть вставлен слайд;
- d) #Главная# - !Буфер обмена! – [Вставить].

#### **5.5. Сортировка слайдов**

При сортировке можно изменить порядок следования слайдов в презентации.

Для выполнения сортировки нужно:

- a) пометить перемещаемый слайд
- b) отбуксировать слайд при нажатой левой кнопке мыши в нужное место.

#### **5.6. Редактирование презентации**

Просмотреть содержание всей презентации и внести необходимые корректировки, включая:

- изменения в тексте;
- замену графических объектов;
- изменения порядка следования слайдов в презентации

можно в режиме «Обычный». Для перехода в режим «Обычный»:

#Вид# - !Режимы просмотра презентаций! - [Обычный].

Изменение порядка следования слайдов можно производить путем буксировки. В этом режиме можно также "скрыть" те слайды, которые не

войдут в презентацию, но могут быть вызваны с помощью кнопок управления. Для того, чтобы скрыть слайд, нужно пометить его, вызвать контекстное меню правой кнопкой мыши и выбрать пункт «Скрыть слайд».

### **5.7. Создание и настройка анимации**

Любой объект (надпись, кнопка управления, диаграмма, рисунок) может быть изображен во время демонстрации с эффектами анимации. Для создания эффекта анимации в режиме «Обычный» нужно выделить объект, затем:

- a) #Анимация# - !Анимация! – [Настройка анимации];
- b) выбрать способ анимации во всплывающей панели «Настройка анимации».

### **5.8. Установка режима просмотра презентаций**

Смена слайдов при демонстрации презентации может производиться пользователем (клавишами PageDown и PageUp) или автоматически, по времени. Для установки режима просмотра по времени:

- a) #Показ слайдов# - !Настройка! - [Настройка демонстрации];
- b) во всплывающем диалоговом окне «Настройка презентации» установить опции:
  - автоматически (полный экран);
  - смена слайдов по времени.
- c) #Показ слайдов# - !Настройка! - [Настройка демонстрации].

При этом автоматически будет запущена пробная демонстрация презентации. В процессе пробной демонстрации переход к следующему слайду производится клавишей <PageDown> или <SpaceBar>, а в конце пробной демонстрации пользователю предъявляется лист со слайдами с указанием фактического времени демонстрации каждого.

Если установлен автоматический режим просмотра слайдов, то можно сделать просмотр циклическим и непрерывным до момента нажатия клавиши <Esc>. Для этого нужно:

- a) вызвать панель «Настройка презентации»:  
#Показ слайдов# - !Настройка! - [Настройка демонстрации];

- b) установить флајжок «непрерывный цикл до нажатия клавиши Esc».

### **5.9. Просмотр презентаций**

Показ слайдов при просмотре презентации может производиться в одном из трех режимов:

- a) с первого слайда;
- b) с текущего слайда;
- c) произвольный показ. В этом режиме отображаются только выбранные слайды.

Для начала просмотра:

#Показ слайдов# - !Начать показ слайдов! – [с первого слайда/с текущего слайда/произвольный показ...]

В режиме ручного просмотра переход к показу следующего слайда - клавишей <PageDown> или щелчком левой кнопки мыши, предыдущего - <PageUp>. В автоматическом режиме переход от слайда к слайду - по времени. Выход из режима просмотра - <Esc>.

Переключиться в режим просмотра презентаций можно и в процессе создания презентации.

### **5.10. Изменение последовательности просмотра слайдов**

- a) #Показ слайдов# - !Начать показ слайдов! – [произвольный показ...];
  - b) во всплывающем диалоговом окне «Произвольный показ» сделать щелчок левой кнопкой мыши на кнопке «Создать». В результате этого появится диалоговое окно «Задание произвольного показа»;
  - c) в диалоговом окне «Задание произвольного показа», используя элементы управления окно «Слайды презентации» и кнопки «Добавить» и «Удалить» создать список слайдов в окне «Слайды произвольного показа». После завершения сделать щелчок левой кнопкой мыши на кнопке «OK»;
  - d) во всплывающем диалоговом окне «Произвольный показ» сделать щелчок левой кнопкой мыши на кнопке «Закрыть». Можно произвести пробный просмотр созданного произвольного показа сделав щелчок левой кнопкой мыши на кнопке «Показать».
- б) сформировать последовательный список слайдов в нужном порядке.

### **5.11. Создание и использование скрытых слайдов**

В составе презентации кроме основных слайдов можно предусмотреть дополнительные, которые не входят в состав презентации, а нужны, например, для ответов на вопросы. Эти слайды делаются скрытыми, вызов их выполняется с помощью кнопок управления на «открытых» слайдах. Для того чтобы сделать слайд скрытым, нужно:

- a) открыть презентацию в режиме «Обычный»;
- b) пометить слайд;
- c) #Показ слайдов# - !Настройка! – [Скрыть слайд].

### **5.12. Создание и использование при демонстрации презентации кнопок управления**

PowerPoint предоставляет возможность создавать на слайдах кнопки управления (Action Button). Используя эти кнопки, пользователь может изменять последовательность просмотра и даже переключаться на просмотр других презентаций.

Для создания кнопки управления нужно:

- a) открыть слайд в режиме «Обычный»;
- b) #Вставка# - !Иллюстрации! – [Фигуры] – [Управляющие кнопки] – [<вид кнопки>];
- c) связать кнопку с определенным действием (см. рис. 5.2).

Назначение основных кнопок управления:

**Домой** - например, переход к слайду, содержащему меню, определяющему направление дальнейшего просмотра презентации;

**Справка** - например, вызов тестового файла Word, содержащего справку;

**Сведения** - например, вызов определенного тестового файла Word;

**Назад** - переход к просмотру предыдущего слайда;

**Далее** - переход к просмотру следующего слайда;

**В начало** - переход к просмотру первого слайда;

**В конец** - переход к просмотру последнего слайда;

**Возврат** - возврат из режима просмотра слайдов в режим редактирования слайдов;

**Документ** - например, вызов определенного тестового файла Word;

**Звук** - переход к звуковому клипу;

**Фильм** - переход к видеоклипу;

**Настраиваемая** - может быть настроена на любую из указанных выше функций.

Примечание: точно так же любую кнопку можно настроить на любую функцию. Но на всех кнопках, кроме настраиваемой, имеются пиктограммы, соответствующие названным функциям.

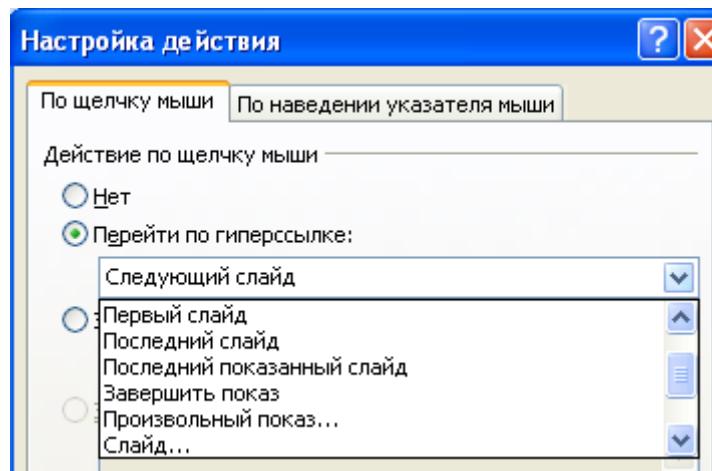


Рис. 5.2.

Использование кнопок управления дает возможность построить иерархическую систему показа с разветвлениями. С помощью кнопок управления можно также производить переход к "скрытым" слайдам.

После создания кнопки управления можно задать также действие запуска какого-либо exe-модуля (например, Turbo.exe). Для этого в диалоговом окне «Настройка действия» нужно активировать пункт «Запуск программы» и, сделав щелчок левой кнопкой мыши на кнопке «Обзор...», найти нужный exe-модуль во всплывающем диалоговом окне «Выберите программу для запуска»

### 5.13. Добавление к презентации раздаточного материала

Раздаточный материал представляет собой уменьшенные изображения слайдов (от 2 до 6 слайдов на странице). На листе могут быть текстовые комментарии.

Образец (макет) листа выдач можно посмотреть. Для этого:  
#Вид# - !Режимы просмотра презентации! - [Образец выдач].

На листы выдач по умолчанию добавляются верхний и нижний колонтитулы, в колонтитулах отображаются номер страницы, дата и время. Для корректировки содержимого колонтитулов нужно выполнить следующую последовательность действий:

- a) #Вид# - !Режимы просмотра презентации! - [Образец выдач];
- b) #Образец выдач# - [Заполнители];
- c) снять/установить флажки:
  - верхний колонтитул;
  - нижний колонтитул;
  - дата;
  - номер страницы.

Для настройки на печать выдач:

- a) OFFICE - [Печать...] – [Предварительный просмотр];
- b) во всплывающем окне «Просмотр» на вкладке «Предварительный просмотр» в группе «Параметры страницы» в поле со списком «Печатать следующее» выбрать режим печати выдач.

## **Упаковка презентаций**

В результате упаковки будет сформирован файл презентации, который может быть записан на компакт-диск локальный или сетевой диск. Также копируются приложение Microsoft Office PowerPoint Viewer 2016 и все связанные файлы (например, фильмы или звуки).

Упакованный файл презентации может быть запущен на этом или другом компьютере с установленной операционной системой Windows 10/8/7/XP. Для просмотра презентации, представленной в упакованном виде, наличие программы Microsoft PowerPoint не требуется. В упакованном файле презентации будет содержаться PowerPoint Viewer.

Примечание.

- Копирование презентации возможно на чистый записываемый (CD-R) или перезаписываемый (CD-RW) компакт-диск, а также на CD-RW-диск с содержимым, которое при этом будет стерто.
- В приложении PowerPoint перед копированием презентации на компакт-диск убедитесь, что за один сеанс записи можно скопировать все необходимые файлы. По окончании копирования

## Список литературы

*первого набора файлов добавить файлы на диск с помощью приложения PowerPoint будет невозможно. Однако для копирования дополнительных файлов на CD-R-диск или CD-RW-диск, содержащий существующие файлы, можно использовать проводник Windows.*

Для выполнения упаковки нужно выполнить следующую последовательность действий:

- a) OFFICE - [Опубликовать] – [Подготовить для компакт-диска];
- b) в диалоговом окне «Упаковка для записи» в поле «Название компакт-диска» выбрать имя компакт-диска или папки, в которую нужно скопировать презентацию.

## Список литературы

1. Joan Lambert and Curtis Frue. Microsoft Office 2016/ Step by Step. Microsoft Press, 2015 – 564 р.
2. Леонтьев В. Office 2016. Новейший самоучитель.:Изд-во «Эксмо», 2015 -368 с.
3. Бекаревич Ю. Б., Пушкина Н. В. Самоучитель Microsoft Access 2010. СПб.: БХВ-Петербург, 2011. -752 с
4. Гарнаев А. Самоучитель VBA. СПб.: ВHV-Санкт-Петербург, 2002 - 512 с.
5. Гарнаев А.Ю. VBA. – СПб.: БХВ – Петербург, 2005. – 848 с.