

# Оглавление

## Введение

9

## 1

### Введение в офисное программирование

11

Цель разработки	12
Область применения	12
Язык программирования	12
Среда разработки	13
Поддержка ООП	13
Преимущества офисного программирования	13

## 2

### Макросы. Использование макрорекордера

15

Назначение макросов	16
Запись макросов. Использование макрорекордера	17
Воспроизведение макроса	19
Редактирование макроса	26

## 3

### Интерфейс редактора Visual Basic

47

Строка меню	48
Инструментальные панели	49
Окно проектов	50
Форма	51
Создание экранных форм VBA	52
Дополнительные элементы управления	101
Краткое описание дополнительных элементов управления	116
Вставка элементов управления на форму	119
Окно кода	121
Окно выполнения	122
Панель элементов управления	122
Настройка редактора VBA	123

## 4

### Синтаксис VBA

131

Общие положения	132
Типы данных	136
Объявление переменных	140
Локальные переменные	151

Объявление статических переменных	152
Объявление констант	154
Создание пользовательских типов	158
Преобразования типов	158
Верхние и нижние колонтитулы	160
Диапазоны в VBA	162
Выражения VBA	169
Операторы VBA	170
Свойства объектов	185
Свойства книги	188
Свойства листа	189
Свойства диаграммы	193
Свойства формы	194
Свойства элементов управления	196
Примеры создания экранных форм	202
Режим конструктора	214
Функция диалога MsgBox	219
Функция InputBox	227
Метод InputBox	231
Выравнивание элементов управления на форме	235
Практическая работа № 1. Преобразование типов данных	239
Практическая работа № 2. Программирование линейных и разветвляющихся алгоритмов	241

## 5

### Организация циклов

245

Оператор цикла For-Next	246
Оператор цикла Do-While	248
Оператор цикла Do-Until	250
Конструкция For Each-Next	251
Практическая работа № 3. Программирование циклических вычислительных процессов	253

## 6

### Пользовательские подпрограммы в VBA

255

Функция	256
Процедуры-функции	258
Аргументы функций	264
Процедуры VBA	265
Вызов процедуры	271
Аргументы процедур	277
Процедуры свойств	280
Практическая работа № 4. Программирование с использованием функций	283

## 7

**Модули VBA****287**

Типы модулей: модули форм, стандартные модули, модули классов	288
Области видимости переменных	289
Модульные переменные	289
Глобальные переменные	290

## 8

**Структурные типы данных****293**

Объявление массивов	294
Практическая работа № 5. Программирование циклических вычислительных процессов с использованием массивов	304
Практическая работа № 6. Программирование с использованием составных пользовательских типов данных	306

## 9

**Объектная модель компонентов Microsoft Office****309**

Коллекции объектов в VBA	310
Свойства объектов	311

## 10

**Создание форм. Обработка событий****315**

События VBA	316
Блокировка и разблокировка событий	426
Практическая работа № 7. Разработка пользовательских диалоговых окон (форм)	428

## 11

**Интеграция с внешними приложениями****433**

Основы автоматизации	434
Ссылка на библиотеку объектов приложения-сервера	434
Отладка	436

**Заключение****441****Список литературы****443**

# Введение

Microsoft Excel — электронные таблицы, которые предназначены для вычислений, анализа и построения разнообразных отчетов.

Сегодня этот программный продукт — абсолютный лидер среди аналогичных программ электронных таблиц по своей простоте и вместе с тем колоссальным возможностям.

Электронные таблицы Microsoft Excel разработаны компанией Microsoft и обновляются примерно раз в два года. В каждой новой версии Microsoft Excel расширяются возможности этой программы. Все, чему пользователь учится в одной программе, он найдет и в следующей версии, плюс что-то новое.

Даже самая совершенная программа не может охватить всего многообразия реальных ситуаций и задач. Чтобы вы могли самостоятельно решать такие задачи, в программе Microsoft Excel существуют макросы.

Как правило, в учебниках по программе Microsoft Excel об этом понятии говорится вскользь, и это неудивительно: к изучению макросов пользователи подходят уже подготовленными специалистами по Microsoft Excel. Чтобы создавать макросы, нужно видеть проблему, которая решается нестандартным способом.

Умение самостоятельно решать такие проблемы позволит вам не только почувствовать себя крутым специалистом, которых компании зазывают к себе на работу, но и самой организации сэкономят много финансовых средств, которые были бы потрачены на поиск программистов.

Программирование в среде Excel осуществляется на языке Visual Basic for Applications.

Слово *Basic* — это аббревиатура (Beginners Allpurpose Symbolic Instruction Code).

1

# Введение в офисное программирование

## Цель разработки

Офисное программирование предназначено для разработки приложений, используемых для автоматизации офисной деятельности. У приложений различное назначение: Microsoft Office, OpenOffice, LibreOffice, CorelDRAW Graphics Suite.

Цель — совершенствовать работу с документами под требования конкретного пользователя. Бывают макросы постоянного и разового применения. Вопрос в том, как выгоднее и быстрее обработать информацию: с помощью макроса или вручную.

## Область применения

Макрос входит в состав документов: или одного, или встраивается в шаблон документа. В последнем случае макрос встраивается в каждый создаваемый документ. Макрос может изменять настройки документа или использоваться для обработки других документов. Может, например, изменять параметры шрифта, ориентацию страниц, создавать разделы, изменять ориентацию страниц в каждом из разделов, исправлять одни буквы на другие — *і* на *и*, *ѣ* на *е*, *ѿ* на *ф* в дореволюционных книгах и т. д. Как самостоятельная программа — отдельно от документа, рабочей книги, презентации — макрос не используется.

## Язык программирования

В офисном программировании, как правило, используется язык Visual Basic for Applications (VBA). В некоторых приложениях (OpenOffice, LibreOffice, CorelDRAW Graphics Suite) используется версия языка StarBasic — в связи с другим набором объектов.

Язык Visual Basic for Applications — специализированная версия языка Visual Basic.

## Среда разработки

Среда разработки — приложения Microsoft Office. Каждое имеет свой набор объектов: Документ ((Documents) в Word, Рабочий лист (WorkSheets) и Рабочая книга (Workbook) в Excel) и т. д.

Макросы создаются двумя способами: включением макрорекордера и ручным программированием. Макрорекордер записывает все действия, а программные строки записываются автоматически. Запись макроса вручную выполняется с помощью специального редактора, который загружается из риббона Разработчик.

## Поддержка ООП

Язык VBA поддерживает объектно-ориентированное программирование. Документы, листы, книги, презентации, методы и события — это объекты. Например, многие цвета объектов имеют свои имена и/или константы, по ним эти цвета можно вызвать. Каждый элемент на форме и сама форма тоже объекты, у них есть свойства, которые можно изменить.

Объектно-ориентированное программирование — основа визуального программирования, то есть свойства объектов можно изменять не только в тексте программы, но и с помощью специальной панели. Но эта возможность может повлечь ошибки у неопытных пользователей. Поэтому многие преподаватели не разрешают студентам в начале обучения изменять свойства объектов на панели свойств, а использовать их только при обработке события инициализации формы.

## Преимущества офисного программирования

Возможность использовать макрорекордер позволяет пользователю изменять документ, не привлекая программистов.

Пользователь может продолжать работать в привычном приложении, получая новые возможности.

Многие приложения (OpenOffice, LibreOffice, Mozilla Firefox) могут изменять возможности с помощью расширений и дополнений, их разрабатывают сторонние программисты. Для этих приложений организованы специальные интернет-магазины и библиотеки, можно скачать и установить такие дополнения.



2

## Макросы. Использование макрорекордера

## Назначение макросов

Макрос — это последовательность команд, которые выполняются в Microsoft Office, OpenOffice, LibreOffice, Corel Graphic Suite. В этих приложениях макросы можно создавать на языках Visual Basic for Applications, его версии StarBasic, многих других. Макросы — самостоятельные программные продукты, но способны выполняться только внутри одного из приложений. Они могут быть потенциально опасными, поэтому о существовании макросов обычно предупреждают в устаревших форматах: DOC, XLS и др. В новейших форматах (DOCX, XLSX) макросы сохранить невозможно: для этого предусмотрены специальные форматы (DOCM, XSLM). Буква *M* в формате говорит о специализации формата для хранения макросов. Такое разделение форматов по назначению в некоторой степени защищает пользователей от макровирусов при скачивании файлов из интернета.

Во многих приложениях для создания макросов используется язык Visual Basic for Applications (VBA) или его версии (StarBasic), но могут использоваться и другие языки: Python, BeanShell, JavaScript, Lisp (если это предусмотрено). Во многих приложениях, хотя об этом и не сообщается явно, используется язык запросов SQL.

Макросы нужны для автоматизации часто повторяющихся операций. На практике наибольшее применение макросов встречается в электронных таблицах. Поэтому в примерах будут использованы именно электронные таблицы (на примере Microsoft Excel). Рассмотрим несколько простейших примеров.

1. Отделу кадров часто приходится вводить информацию с персональными данными сотрудников: фамилию, имя, отчество. Для выбора ширины столбца можно разработать макрос.
2. В электронных таблицах нужно форматировать заголовки, шапку таблицы: размер шрифта, начертание и т. д.
3. Оформление бланков и отчетов. В столбцах можно создать формулы. Пользователю остается заполнить бланк или отчет числами, по которым немедленно будут сделаны расчеты.

4. Электронная библиотека Максима Мошкова предлагает множество дореволюционных книг. С помощью макрорекордера можно отредактировать текст в современном написании (без ятей и ижиц) и сделать книги удобными для чтения и конвертирования в электронные форматы (FB2, EPUB, PDF, DjVu).

Для создания макросов есть два способа: автоматическое создание команд по принципу магнитофона (авторекордер) и ручное создание в редакторе Visual Basic for Applications. Оба метода взаимосвязаны: в любой момент один может дополнить другой. Например, можно записать некоторую последовательность команд с помощью макрорекордера, а затем отредактировать полученный текст программы в редакторе VBA.

## Запись макросов. Использование макрорекордера

Создадим пример. Введем пять записей с ФИО в трех столбцах (строку 7 не вводите), а затем включим макрорекордер, и для каждого столбца будет определена оптимальная ширина (рис. 1).

	A	B	C	D		A	B	C
1	Фамилия	Имя	Отчество		1	Фамилия	Имя	Отчество
2	Амирова	Дарья	Руслановна		2	Амирова	Дарья	Руслановна
3	Гаджимур	Амина	Байрамбековна		3	Гаджимурадова	Амина	Байрамбековна
4	Копытина	Екатерин	Дмитриевна		4	Копытина	Екатерина	Дмитриевна
5	Айнуллин	Рушан	Раисович		5	Айнуллин	Рушан	Раисович
6	Коловато	Олеся	Александровна		6	Коловатова	Олеся	Александровна
7	Байрамов	Тельман	Бюньямутдинович		7	Байрамов	Тельман	Бюньямутдинович

Рисунок 1. Столбцы до и после применения макроса

Алгоритм создания макроса: выполните команду **Разработчик** ⇒ **Запись макроса**: укажите имя нового макроса (рис. 2), создайте описание назначения макроса, нажмите на кнопку **ОК**.

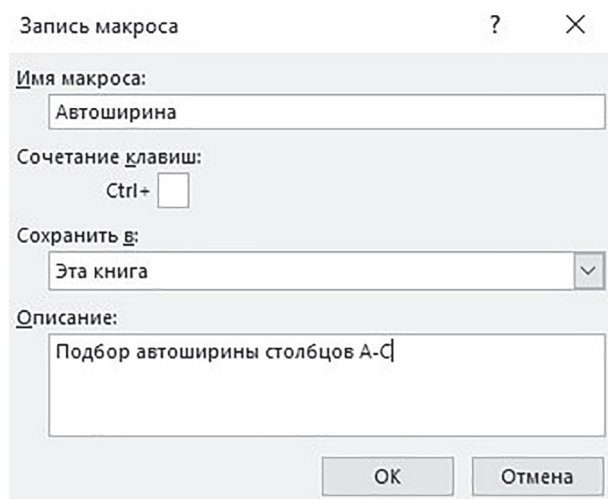


Рисунок 2. Диалоговое окно Запись макроса

Установите автоширину каждого из трех столбцов двойным щелчком между заголовками столбцов, выполните команду **Разработчик** ⇒ **Остановить запись**. Макрос готов, и его программный код можно просмотреть командой **Разработчик** ⇒ **Макрос** ⇒ **Изменить**. Появится следующий программный код:

```
Sub Автоширина ()
'
' Автоширина Макрос
' Подбор автоширины столбцов
'
'
Columns("A:A").EntireColumn.AutoFit
Columns("B:B").EntireColumn.AutoFit
Columns("C:C").EntireColumn.AutoFit
End Sub
```

Команда (AutoFit) устанавливает автоширину (Columns) для указанного столбца. Обратите внимание на имена ячеек (A:A, B:B, C:C) — автоширина подбирается по всему столбцу.

Обратите внимание на двойные компьютерные кавычки ("" ) — в тексте программ допускаются только такие кавычки. Другие типы кавычек (« », “ ”) не допускаются. Распространенная ошибка среди начинающих программистов — вместо одной двойной кавычки часто вводят две одиночные кавычки, найти такую ошибку очень сложно. Такие ошибки при использовании макрорекордера допущены не будут, так как он знает, какие кавычки допустимы, а вот при ручном написании текста программ ошибки с кавычками могут быть. Часто недопустимые кавычки оказываются в тексте программ при:

- 1) копировании текста программы из интернета;
- 2) при сканировании и оптическом распознавании текста, а потом при копировании этого текста и его вставке в свою программу.

## Воспроизведение макроса

Добавьте новую строку с длинными ФИО. В примере (рис. 1) фамилия и отчество нового работника (строка 7) не умещаются по ширине столбцов. Выделите ячейку A1.

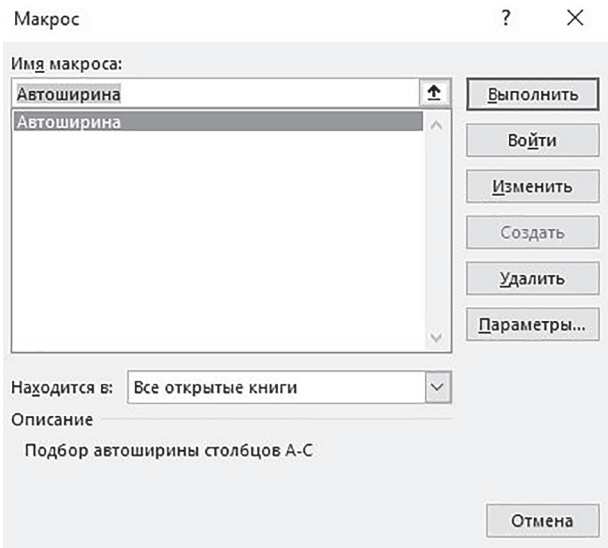


Рисунок 3. Выбор макроса

Для воспроизведения макроса выполните команду **Разработчик** ⇒ **Макрос** ⇒ **Выполнить**. В открывшемся диалоговом окне (рис. 3) выберите имя макроса, который необходимо воспроизвести, и щелкните по нему левой клавишей. После выбора макроса в списке он появляется в поле **Имя макроса**. Чтобы воспроизвести выбранный макрос, необходимо нажать на кнопку **Выполнить**.

### Макрос: создание заголовка таблицы

Рассмотрим другой пример, в котором в ячейке A5 вводится заголовок таблицы. Создадим макрос, который должен выполнять следующие действия:

- вводить текст *Отчет за II квартал*;
- устанавливать размер шрифта в 20 пунктов;
- форматировать текст полужирным и курсивным начертанием;
- устанавливать автоширину столбца (рис. 4).

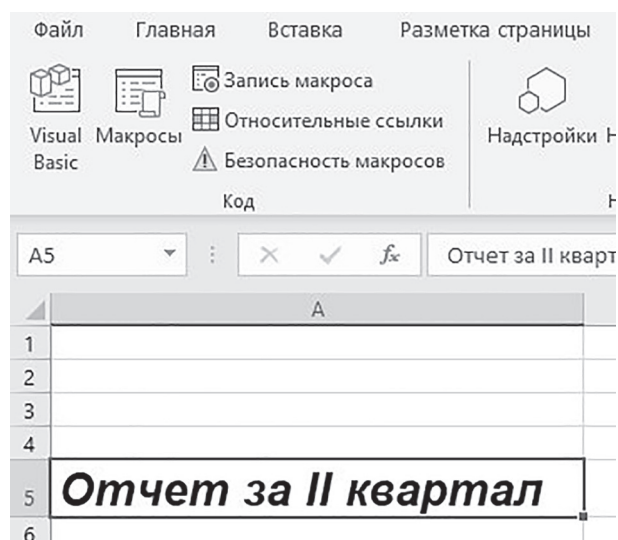


Рисунок 4. Оформленный заголовок

Алгоритм создания макроса в этом примере ничем не отличается от предыдущего: выполняем команду **Разработчик** ⇒ **Макрос** ⇒ **Начать запись**, задаем имя новому макросу,

вводим неформатированный текст, форматируем ячейку полужирным и курсивным начертанием, устанавливаем автоширину столбца, выполняем команду **Разработчик** ⇒ **Макрос** ⇒ **Остановить запись**. Макрос готов. Его программный код можно просмотреть командой **Разработчик** ⇒ **Макрос** ⇒ **Макросы** и в открывшемся диалоговом окне **Макрос** нажать на кнопку **Изменить**. Появится следующий программный код:

```
Sub Макрос2 ()
'
' Макрос2 Макрос
' Макрос записан 25.11.2022 (Шитов)
'
'
'
Columns("A:A").EntireColumn.AutoFit
Range("A5").Select
ActiveCell.FormulaR1C1 = "Отчет за II квартал"
With ActiveCell.Characters(Start:=1,
Length:=19).Font
    .Name = "Arial Cyr"
    .FontStyle = "полужирный курсив"
    .Size = 20
    .Strikethrough = False
    .Superscript = False
    .Subscript = False
    .OutlineFont = False
    .Shadow = False
    .Underline = xlUnderlineStyleNone
    .ColorIndex = xlAutomatic
End With
Rows("5:5").Select
End Sub
```

Из всего программного кода отметим только значения булевого типа, которые могут принимать следующие значения: или *True* (*истина*), или *False* (*ложь*). Опция *True* означает, что данный параметр включен, а опция *False* — параметр выключен.