# Практическая работа №5 Знакомство с системой VBA.Структура редактора VBA

## Тема и цель работы

Знакомство с интерфейсом редактора VBA. Рассмотрение структуры кода на примере кода простого макроса MS Excel.

## Запуск редактора VBA

Все приложения Office 2010 используют ленту. Одной из вкладок на ленте является вкладка **Разработчик** (рис. 4), на которой можно вызвать редактор Visual Basic и другие инструменты разработчика. Поскольку в Office 2010 вкладка **Разработчик** не показана по умолчанию, необходимо вывести ее на экран, выполнив следующую процедуру.

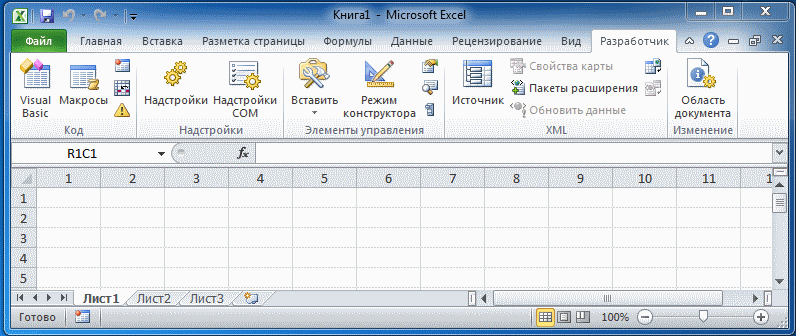


Рис. 4 Вкладка **Разработчик** на ленте в Excel

На вкладке **Файл** выбрать **Параметры**, чтобы открыть диалоговое окно **Параметры Excel**. Щелкнуть **Настройка ленты** в левой части диалогового окна. В разделе **Выбрать команды**, расположенном слева в окне, выбрать **Часто используемые команды**. В разделе **Настройка ленты**, который находится справа в диалоговом окне, выбрать **Основные вкладки**, а затем установить флажок **Разработчик**.

Также Редактор VBA можно открыть при помощи сочетания клавиш **<ALT+F11>**. Возвратиться из редактора VBA в рабочую книгу можно нажатием кнопки Вид–Microsoft Excel **(View–Microsoft Excel)**. Интерфейс VBA состоит из следующих основных компонентов: окно проекта, окно свойств, окно редактирования кода, окна форм, меню и панели инструментов (рис. 5).

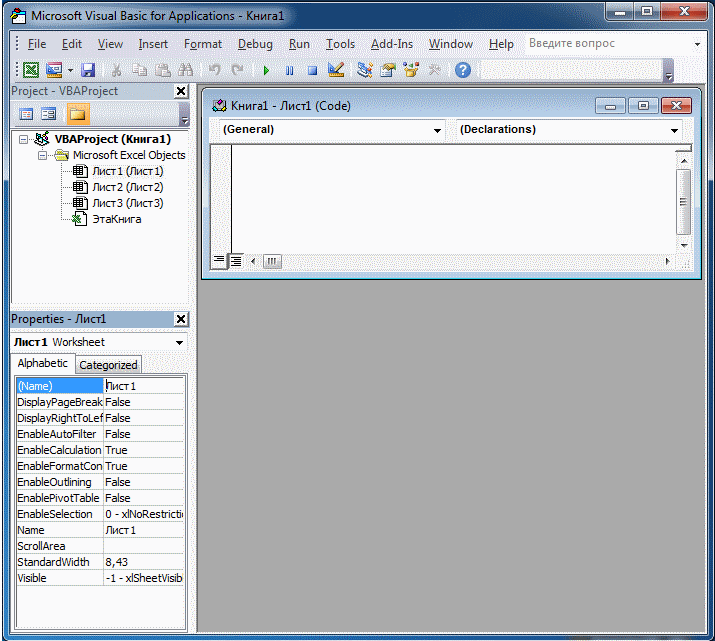


Рис. 5 Редактор VBA

## Проблемы безопасности

Нажмите кнопку **Безопасность макросов**, чтобы определить, какие макросы могут выполняться и при каких условиях. Хотя неконтролируемый код макроса может серьезно повредить компьютер, условия безопасности, запрещающие выполнять полезные макросы, могут серьезно ухудшить производительность работы. Безопасность макросов – это сложный и важный вопрос, в котором следует разобраться при работе с макросами Excel.

Необходимо помнить, что если при открытии книги, содержащей макрос, между лентой и листом появляется строка **Предупреждение системы безопасности: запуск макросов отключен**, можно нажать кнопку **Включить содержимое**, чтобы включить макрос.

Кроме того, в качестве мер безопасности, нельзя сохранить макрос в формате файлов Excel, используемом по умолчанию (XLSX-файлы), вместо этого макрос должен быть сохранен в файл со специальным расширением, XLSM-файл.

## Окно проекта (Project)

Окно проекта в редакторе VBA активизируется выбором команды **Вид, Окно проекта** (View, Projectexplorer). В окне проекта (Project – VBAProject) представлена иерархическая структура файлов форм и модулей текущего проекта (рис. 6).

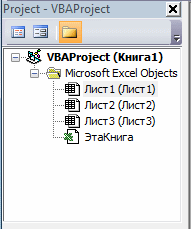


Рис. 6 Окно проекта

В проекте автоматически создается модуль для каждого рабочего листа и для всей книги. Кроме того, модули создаются для каждой пользовательской формы, макросов и классов. По своему предназначению модули делятся на два типа: модули объектов и стандартные. К стандартным модулям относятся те, которые содержат макросы. Такие модули добавляются в проект командой **Вставка –Модуль(Insert–Module).** К модулям объектов относятся модули, связанные с рабочей книгой, рабочими листами, формами, и модули класса.

Формы создаются командой Вставка –UserForm **(Insert–UserForm),** а модули класса – командой Вставка –Модуль класса**(Insert–Module).** По мере создания, добавления и удаления файлов из проекта эти изменения отображаются в окне проекта. Отметим, что удаление файла из окна проекта производится выбором значка файла с последующим выполнением команды Файл –Удалить**(File–Delete).**

В окне проекта выводится проект всех открытых рабочих книг. Это позволяет легко копировать формы и коды из одного проекта в другой, что убыстряет процесс создания новых приложений.

## Окно свойств (Properties)

В окне свойств отображаются основные параметры свойств выбранной формы или элемента управления (рис. 7). Для отображения этого окна надо либо щелкнуть кнопку **Properties Window**, либо выбрать команду **View – Properties Window**, либо нажать клавишу <**F4**>.

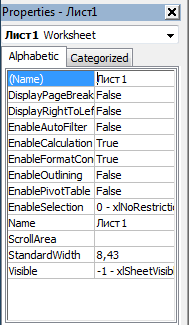


Рис. 7 Окно свойств

В верхней части окна свойств можно из раскрывающегося списка выбрать как форму, так и любой элемент на ней. Основная часть окна свойств состоит из списка свойств выбранного элемента, каждое из которых можно изменить, при необходимости, или с клавиатуры, или выбрать одно из значений выбранного свойства из раскрывающегося списка.

## Окно редактирования кода

Окно редактирования кода (рис. 8) служит в качестве редактора для ввода и изменения кода процедур приложения. Код внутри модуля организован в виде отдельных разделов для каждого объекта, программируемого в модуле.

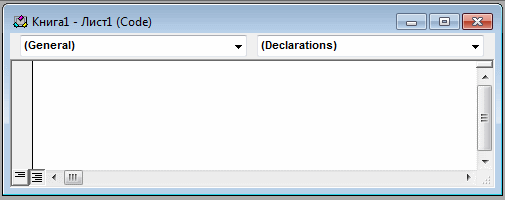


Рис. 8 Окно редактирования кода

В окне редактирования доступны два режима представления кода: просмотр отдельной процедуры и всего модуля. Переключение режимов работы окна редактирования кода осуществляется выбором одной из двух кнопок в нижнем левом углу окна редактирования кода, либо установкой или снятием флажка Просмотр всего модуля **(Default to Full Module View)** вкладки Редактор **(Editor)** диалогового окна Параметры **(Options),** отображаемого на экране командой **Инструменты–**Параметры **(Tools–Options)**.

Два раскрывающихся списка в верхней части окна редактора кода облегчают ориентацию в процедурах. Левый раскрывающийся список позволяет выбрать управляющий элемент или форму, а правый – содержит список событий, допустимых для выбранного в левом списке объекта. Отметим, что при выборе элемента управления в форме посредством двойного щелчка или перемещении указателя на элемент управления и нажатии кнопки Программа **(View Code)** открывается окно редактирования кода как раз в том месте, где располагается процедура, связанная с этим элементом управления. Обратный переход от процедуры к объекту управления быстрее всего осуществить нажатием кнопки Объект **(View Object).**

## Интеллектуальные возможности редактора кода

Написание программ существенно облегчается за счет способности редактора кода автоматически завершать написание операторов, свойств и параметров. При написании кода редактор сам предлагает пользователю список компонентов, логически завершающих вводимую пользователем инструкцию. Например, набирая код Range("A1") после ввода точки на экране отобразится список компонентов (рис. 9), которые логически завершают данную инструкцию. Двойной щелчок на выбранном элементе из этого списка или нажатие клавиши <**Tab**> вставляет выбранное имя в код программы.

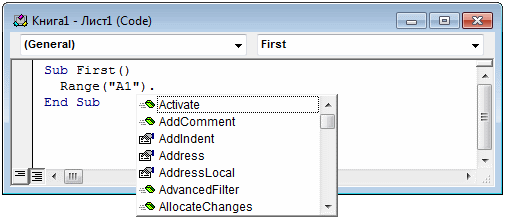


Рис. 9 Список компонентов

Автоматическое отображение списка компонентов происходит только при установленном флажке Список компонентов **(Auto List Members)** вкладки Редактор **(Editor)** диалогового окна Параметры **(Options),** отображаемого на экране после выбора команды **Инструменты** –Параметры **(Tools–Options).**

Список компонентов можно выводить на экран нажатием комбинации клавиш <**Ctrl+J**>, при этом список отображается как при установленном, так и при снятом флажке Список компонентов **(Auto List Members)** вкладки Редактор **(Editor)** диалогового окна Параметры **(Options).**

Отображение списка компонентов, логически завершающих вводимую инструкцию, является одним из интеллектуальных качеств редактора кода. Этим качеством интеллектуальные ресурсы редактора кода не исчерпываются. Другим его такого рода качеством является автоматическое отображение на экране сведений о процедурах, функциях, свойствах и методах после набора их имени (рис. 10).

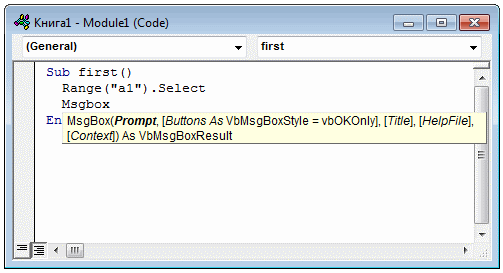


Рис. 10 Отображаемые сведения о вводимой процедуре

Автоматическое отображение на экране сведений о процедурах, функциях, свойствах и методах после ввода их имени происходит только при установленном флажке Краткие сведения **(Auto Quick Info)** вкладки Редактор **(Editor)** диалогового окна Параметры **(Options),** отображаемого на экране после выбора команды **Инструменты**–Параметры **(Tools–Options)**.

Описанную выше всплывающую подсказку можно также выводить на экран нажатием комбинации клавиш <**Ctrl+I**>. При этом всплывающая подсказка отображается как при установленном, так и при снятом флажке Краткие сведения вкладки Редактор диалогового окна Параметры **(Options).**

Редактор кода также производит автоматическую проверку синтаксиса набранной строки кода сразу после нажатия клавиши <**Enter**>. Если после набора строки и нажатия клавиши <**Enter**> строка выделяется красным цветом, то это как раз и указывает на наличие синтаксической ошибки в набранной строке. Эту ошибку необходимо найти и исправить. Кроме того, если установлен флажок Проверка синтаксиса **(Auto Syntax Check)** вкладки Редактор **(Editor)** диалогового окна Параметры **(Options),** отображаемого на экране посредством выбора команды **Инструменты**–Параметры **(Tools–Options)**, помимо выделения красным цветом фрагмента кода с синтаксической ошибкой, на экране отображается диалоговое окно, поясняющее, какая возможная ошибка произошла.

Редактор кода обладает еще одной интеллектуальной возможностью, увеличивающей эффективность работы пользователя. Если курсор расположить на ключевом слове языка VBA, имени процедуры, функции, свойства или метода и нажать клавишу <**F1**>, то на экране появится окно со справочной информацией об этой функции.

## Окно Просмотр объектов (Object Browser)

Окно Просмотр объектов **(Object Browser)** вызывается командой Вид–Просмотр объектов **(View–Object Browser)** или нажатием кнопки Просмотр объектов **(Object Browser)** (рис. 11). В этом окне приведен список всех объектов, которые имеются в системе и которые можно использовать при создании проекта.

Окно Просмотр объектов **(Object Browser)** состоит из трех основных частей:

1. Раскрывающегося списка Проект/Библиотека **(Project/Library)** в левом верхнем углу окна. В этом раскрывающемся списке можно выбрать различные проекты и библиотеки объектов. В частности, библиотеки объектов Excel, VBA, Office и VBAProject (объекты пользовательского проекта).

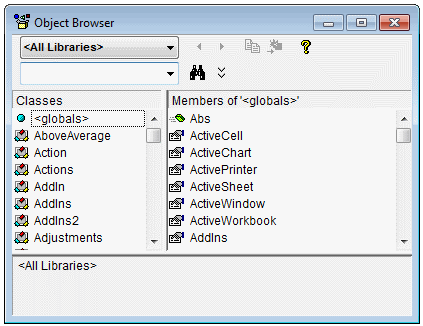


Рис. 11 Окно Просмотр объектов

Выбор в списке строки Все библиотеки **(Аll Libraries)** отображает список объектов всех библиотек.

1. Списка Классы (Classes). После выбора из раскрывающегося списка Проект/Библиотека (Project/Library) просматриваемой библиотеки, например, VBA, все классы объектов выбранной библиотеки выводятся в списке Классы (Classes).
2. Списка Компоненты (Members). После выбора класса из списка Классы (Classes) просматриваемой библиотеки, например, FileSystem, все компоненты выбранного класса выводятся в списке Компоненты (Members). При выделении строки в этом списке в нижней части окна Просмотр объектов (Object Browser) приводится дополнительная информация о выбранном компоненте. Кроме того, если нажать на кнопку Справка (Help), расположенную на панели инструментов в правой верхней часта окна Просмотр объектов (Object Browser), то на экране отобразится окно Справочник Visual Basic (Microsoft Visual Basic Help) с подробной информацией о выделенном компоненте.

## Структура программ на VBA

А теперь подробнее рассмотрим следующие понятия. Программы на VBA хранятся в проектах. Проект содержит модули различных типов, а модули включают различные процедуры.

Проект может содержать несколько модулей. Имеется три типа модулей:

* стандартные модули – это модули, в которых можно описать доступные во всем проекте процедуры;
* модули класса содержат описание объекта, который является членом класса. Процедуры, написанные в модуле класса, используются только в этом модуле. Среди модулей класса выделяют модули форм и отчетов, которые связаны с конкретной формой или отчетом, и Модули форм и отчетов часто содержит процедуры обработки событий, которые срабатывают в ответ на событие в форме или отчете. Процедуры обработки событий используются для управления поведением форм и отчетов и их реакцией на действия пользователя типа щелчка мыши на кнопке.

Модули содержат описания и процедуры – наборы описаний и инструкций, сгруппированных для выполнения. Существует три типа процедур:

* процедураSub – набор команд, с помощью которого можно решить определенную задачу. При ее запуске выполняются команды процедуры, а затем управление передается в приложение пакета MSOffice или процедуру, которая вызвала данную процедуру.
* процедураFunction (функция) также представляет собой набор команд, который решает определенную задачу. Различие заключается в том, что такие процедуры обязательно возвращают значение, тип которого можно описать при создании функции.
* процедураPropertyиспользуется для ссылки на свойство объекта. Данный тип процедур применяется для установки или получения значения пользовательских свойств форм и модулей.

## Окно редактирования форм (UserForm)

Для создания диалоговых окон, разрабатываемых приложений в VBA, используются формы. Редактор форм является одним из основных инструментов визуального программирования. Форма в проект добавляется с помощью команды Вставка–Форма (Insert–UserForm). В результате на экран выводится незаполненная форма с панелью инструментов Панель элементов (Toolbox) (рис. 12).

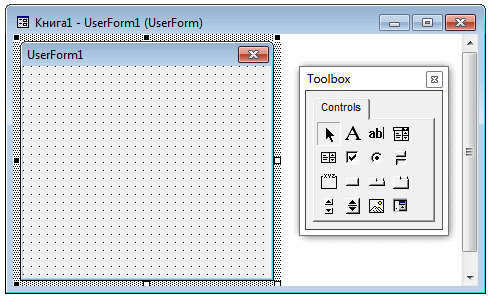


Рис. 12 Окно редактирования форм и панель инструментов

Используя панель инструментов Панель элементов из незаполненной формы, можно сконструировать любое требуемое для приложения диалоговое окно. Размещение нового управляющего элемента в форме осуществляется следующей последовательностью действий:

1. Щелкнуть значок того элемента, который необходимо разместить в форме.
2. Поместить указатель мыши на то место, где будет располагаться управляющий элемент.
3. Нажать левую кнопку мыши и, не отпуская ее, растянуть появившийся прямоугольник до требуемых размеров.
4. Отпустить кнопку мыши. Элемент управления на нужном месте создан.

Размеры формы и расположенных на ней элементов управления можно изменять. Технология изменения размеров стандартная для Windows: выделить изменяемый элемент, поместить указатель мыши на одном из размерных маркеров и протащить его при нажатой левой кнопки мыши так, чтобы объект принял требуемые размеры. Окно редактирования форм поддерживает операции буфера обмена.

Таким образом, можно копировать, вырезать и вставлять элементы управления, расположенные на поверхности формы. Для облегчения размещения и выравнивания элементов управления используется сетка. Активизировать ее можно с помощью вкладки Общие (General) диалогового окна Параметры (Options), вызываемого командой Сервис–Параметры (Tools–Options), там же устанавливается шаг сетки. Кроме того, команды меню Формат (Format) автоматизируют и облегчают процесс выравнивания элементов управления как по их взаимному местоположению, так и по размерам (рис. 13).

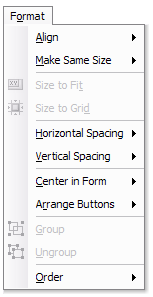


Рис. 13 Команды меню Формат

## Задания на лабораторную работу

Задание 1. Открыть и проанализировать код VBAмакросов, созданных в предыдущей лабораторной работе.

## Задания на самостоятельную работу

Задание 1. Microsoft Excel. Составить таблицу значений синуса и косинуса для чисел от –1,0 до 1,0 с шагом 0,1. Записать макрос, заполняющий формулу только для первой строки. Открыть созданный макрос и добавить строку (заменив соответствующие адреса ячеек):

Selection.AutoFill Destination:=Range("B1:C21"), Type:=xlFillDefault

Выполнить полученный макрос и проанализировать результат. Анализ результата сохранить в файлах макросов в виде комментария.