



Кубанский государственный  
аграрный университет

Факультет прикладной  
информатики

# Разработка приложений в среде MS Office



## Разработка приложений в среде MS Office

Форма контроля

Зачет

Кафедра

Системного анализа и обработки информации (202, 217 эк)

Ведущий преподаватель

**к.э.н., доцент Яхонтова Ирина Михайловна**



## Цель освоения дисциплины «Разработка приложений в среде MS Office»-

изучение студентами основ языка макрокоманд Visual Basic for Application (VBA) с акцентом на решение типовых офисных задач и задач обмена данными, используя визуальный и объектно-ориентированный характер VBA.



Кубанский государственный  
аграрный университет

Факультет прикладной  
информатики

# Лекция 1

## ОСНОВЫ ПРОГРАММИРОВАНИЯ В СРЕДЕ MS OFFICE



- 1. Понятие офисного программирования**
- 2. История программирования на VBA.**
- 3. Возможности VBA, типы VBA-макросов.**



**Офисное программирование** — это процесс разработки приложений, предназначенных для автоматизации офисной деятельности с использованием специализированных пакетов (MS Office, OpenOffice.org или подобных).



## Особенности программирования в среде MS Office:

- 1. Среда разработки.** *Мощная и разнообразная среда приложений Office, в которой можно создавать документы разного типа и работать с ними. Поскольку эта среда ориентирована в первую очередь не на программистов, а на пользователей, то в ней можно создавать документы без всякого программирования. Поэтому программист обычно начинает не на пустом месте, он начинает работать с документами, их заготовками, созданными пользователями. Сама среда представлена в виде объектов, свойства, методы и события которых доступны в языке программирования VBA.*



**2. Совместная работа.** Совместная работа над документами специалиста в некоторой предметной области и программиста-профессионала характерна для офисного программирования и реально может приводить к качественным эффектам, уменьшая время разработки этих документов и улучшая их качество. При совместной работе над документами может быть существенно сокращен типичный для программных продуктов цикл разработки, включающий такие этапы, как создание прототипа системы и его тестирование.



**3. Цели разработки.** Это одна из главных особенностей офисного программирования. Ранее целью программиста было создание приложения, понимаемого как программа, программный проект. Здесь программист является одним из участников создания системы документов. **Документ**, а не программа, становится целью разработки. Программный проект - это лишь часть документа.



**4. Настройка документов.** Слово "Настройка" (*Custom-built*) является одним из ключевых слов в офисном программировании. Действительно, стандартные возможности среды по созданию и работе с документами велики. Однако, возможность настроить стандартный документ Office, сделать его "по заказу", снабдить его новыми функциями, учитывающими специфику решаемой задачи, подогнать его "под себя" это одна из важнейших особенностей офисного программирования. Настройка может быть очень простой и состоять в том, что стандартный документ получает некоторые полезные дополнительные свойства, расширяющие его возможности или внешний вид. Но настройка может быть очень сложной и документ, сделанный по заказу, может ничем не напоминать обычный



5. **Каркас документа.** Для программиста сам Office - это ни что иное, как обычная совокупность библиотек классов. Также как и для других программных сред, библиотеки классов Office представляют собой каркас приложений, или, что может быть точнее с содержательной точки зрения, - **каркас документов (Framework Documents)**. Эти библиотеки классов Office содержат каркасы офисных документов - текстовых документов, документов, основу которых составляют электронные таблицы, презентации, базы данных. Всякий раз, когда создается новый документ, его каркас по умолчанию составляют объекты библиотек, отобранных по умолчанию для построения этого конкретного каркаса документа.



- 6. Язык программирования + Мир объектов.**  
*К программисту, занимающемуся офисным программированием, предъявляются дополнительные требования: помимо языка программирования он должен изучить мир объектов среды. Более того, крайне полезно уметь работать с этими объектами вручную так, как это делают обычные пользователи Office. В остальном поможет мощная справочная система.*



**7. MacroRecorder.** Этот инструмент позволяет создавать программный проект или, по крайней мере, его отдельные компоненты автоматически без программирования. MacroRecorder это транслятор действий, записывающий действия пользователя при работе вручную и транслирующий их в программу на языке VBA, которую затем можно подправить или расширить.



**8. Два крайних случая.** Среди всех офисных документов выделим два типа документов, которые будем условно называть документами типа "обложка" (программный проект, не использующий обращений к объектам, например игры) и документами типа "ручная работа" (документы, создаваемые пользователями без использования VBA). Использование офисного программирования позволит вывести решение различных задач офиса на новый уровень.



## 9. **Microsoft Office - платформа разработчика.**

*Спектр применения офисного программирования широк - от настройки отдельных документов до создания серьезных решений масштаба предприятия. Благодаря интеграции с семейством серверных продуктов Microsoft, интеграции с продуктами третьих фирм, целью разработки становится создание корпоративных приложений, нацеленных на совместную работу в Internet.*



В офисной среде *программный проект неразрывно связан с документом*, хранится как часть документа и не может существовать независимо от него. **Документ, а не программа, является целью разработки.**



Возможность изменить типовой документ, снабдив его *дополнительными функциями* – это одна из важнейших задач офисного программирования.

Для ее решения офисная среда представляет совокупность библиотек классов, которые составляют каркас (Framework) текстовых документов, электронных таблиц, презентаций, баз данных и приложений на основе этих документов.



**Visual Basic for Application (VBA)** - стандартный макроязык пакета Microsoft Office, предназначенный для расширения функциональных возможностей приложения в котором используется.



## С помощью VBA можно:

- создать собственное диалоговое окно и придать ему требуемый внешний вид;
- создать макросы, расширяющие функциональные возможности приложения, в которое встроен VBA;
- изменить меню приложения Microsoft Office;
- управлять другим приложением Microsoft Office или принадлежащими ему данными;
- объединить данные из нескольких приложений Microsoft Office в одном документе;
- автоматически создавать или изменять страницы Web, совместно используя приложения Microsoft Office и VBA.



Среда MS Office предлагает два способа создания программ, отличающихся подходом к процессу: **использование макрорекордера** и **ручное кодирование (на языке VBA)**. Эти подходы ориентированы на разные категории: непосредственно пользователей и программистов соответственно.



*Интегрированная среда разработки* на VBA (Visual Basic Environment, VBE) - встроенное в MS Office средство для написания, тестирования и отладки приложений на VBA. Среда VBE представляет все возможности для создания законченных офисных приложений, включая средства визуального проектирования пользовательского интерфейса.



Офисное программирование позволяет применять на практике идеи компонентного программирования. Компонентный подход предполагает взаимодействие компонентов, создаваемых в разных программных средах, на разных языках, на разных платформах и находящихся на разных машинах. Работа с компонентами (DLL, ActiveX, AddIns, ComAddIns) является неотъемлемой частью офисного программирования.



Язык **Basic** был разработан профессорами Дартмутского колледжа Дж.Кемени и Т. Курцом в **1965** году как средство обучения и работы непрофессиональных программистов.





Его назначение определено в самом названии, которое является аббревиатурой слов Beginner's All purpose Symbolic Instruction Code (многоцелевой язык символических инструкций для начинающих) и при этом в дословном переводе означает «базовый».



В **1991** году под лозунгом «теперь и начинающие программисты могут легко создавать приложения для Windows» появилась первая версия нового инструментального средства Microsoft Visual Basic.



В начале 90-х годов наметилась отчетливая тенденция включения в приложения, предназначенные для конечного пользователя, средства внутреннего программирования, которые должны были решать задачи настройки и адаптации этих пакетов для конкретных условий их применения.



В конце **1993** г. Microsoft объявила о намерении создать на основе VB новую универсальную систему программирования для прикладных программ, которая получила название Visual Basic for Applications (VB для приложений). Естественно, реализацию этого проекта она начала с собственных офисных пакетов.



Первый вариант VBA 1.0 появился в составе MS Office 4.0, но лишь в программах Excel 4.0 и Project 6.0. В других же приложениях - Word 6.0 и Access 2.0 - были собственные варианты Basic. Более того, VBA 1.0 довольно сильно отличался (причем имея ряд существенных преимуществ) от используемой тогда универсальной системы VB 3.0.



Качественный перелом наступил в конце **1996** года с выпуском MS Office 97, в котором была реализована единая среда программирования VBA 5.0, включенная в программы Word, Excel и PowerPoint. Более того, VBA 5.0 использовала тот же самый языковой механизм и среду разработки, что и универсальная система VB 5.0. В состав MS Office **2000** вошла соответственно версия VBA 6.0, которая используется в шести программах - Word, Excel, PowerPoint, Access, Outlook, Frontpage.



Visual Basic for Applications - это не просто язык программирования, а целая технология, которую вполне можно распространить и на другие приложения сторонних производителей, подарив их пользователям возможность создавать программы, работающие в их среде.



## Основные возможности языка Visual Basic:

1. Реализация разработчиком максимально гибкого и удобного интерфейса для своего приложения, который сможет удовлетворить потребности даже самого взыскательного пользователя.
2. Обработка событий мыши и клавиатуры, вывод на экран различных графических изображений.
3. Использование стандартных диалогов.
4. Обработка файлов и каталогов на жестком диске.
5. Доступ к базам данных.



VBA является встроенным языком программирования приложений MS Office, его можно применить в Excel, Access, Word. Особенностью VBA является то, что при создании приложения за основу берется документ: нельзя создать приложения независимо от документа (рабочей книги в Excel, документа в Word).



**Макрос** происходит от английского слова *macro* и означает программный объект, который при работе может разворачиваться в последовательность команд и действий.

То есть это серия команд и функций, хранящихся в модуле *Visual Basic for Applications*, прикрепленном к данной книге (или шаблону). Модуль представляет собой набор объявлений типа переменных и процедур на языке *VBA*.

Макрос (процедуру) можно выполнять всякий раз, когда необходимо выполнить данную последовательность действий.



**Макрорекордер** (рекордер) - это специализированная программа, которая позволяет записывать все действия пользователя в определенной последовательности, учитывая и исправляя по ходу все возникающие ошибки. При запуске макроса, программа строго следует командам, соответствующим записи, что значительно повышает не только скорость, но и точность выполнения поставленных задач.



Макрорекордер представляет возможность создания программного проекта или, по крайней мере, его отдельных компонентов автоматически, без программирования. Для записи и воспроизведения макроса не требуется специальных знаний, поэтому пользователь может самостоятельно создавать программы (макросы), в общем случае даже не представляя себе как они работают.



Для программиста макрорекордер полезен тем, что позволяет создавать фрагменты программы автоматически, тем самым увеличивая скорость разработки и уменьшая время отладки.



На сегодняшний день существует ряд макрорекордеров, написанных на разных языках программирования, которые разработаны для работы в определенных программных продуктах.



Так, к примеру, для приложений Microsoft Office макросы записываются на Visual Basic for Application, который работает только в них. Владая этим языком можно улучшать макросы, которые записаны макрорекордером, и расширять их возможности.



Прежде чем записать макрос, необходимо продумать выполняемые действия, то есть составить алгоритм. Особое внимание уделяется следующим вопросам:

- какие действия должен выполнять макрос;
- какие предварительные действия должны быть выполнены перед запуском макроса (открытие файла, активизация документа или его объекта, выделение фрагмента, режим работы приложения и пр.);
- какие действия необходимо выполнить после завершения работы макроса.



Каждый раз при записи макроса, он сохраняется в модуле, присоединенном к книге. Редактор Visual Basic for Applications позволяет изменять макросы, а также копировать их либо из одного модуля в другой, либо между различными книгами.

Большая часть макроса представляет собой изменение свойств объектов и выполнение методов.



Макросы всегда выполняются в активном документе. Действие некоторых макросов зависит от позиции вставки. Другие могут обращаться только к определенному объекту, например к абзацу или выделенному участку текста.



Макросы бывают двух типов: **внутренние и внешние.**

**Внутренние** внедряются непосредственно в документ или шаблон и могут быть выполнены, только если пользователь работает с данным файлом.



**Внешний** макрос можно применять в документах, созданных на основе разных шаблонов. Если вы пользуетесь большим количеством шаблонов, вносить в каждый из них макрос будет неудобно, так как придется делать при использовании внутреннего макроса.

Внешний макрос можно применять для работы с любыми документами, а также свободно переносить с компьютера на компьютер.



## Безопасность при работе с макросами

Макросы являются не только очень удобным средством для работы с программами пакета Microsoft Office, но и средством перенесения вирусов с одного компьютера на другой, поэтому относитесь к макросам, полученным из неизвестных или непроверенных источников, осторожно. Вирусы, содержащиеся в макросах, называются макровирусами.

Офисные программы определяют наличие макросов в документе и сообщает вам об этом. Для этого необходимо установить соответствующий уровень доверия макросам на кнопке Безопасность макросов в группе Код на вкладке Разработчик ленты.