Visual Basic for Applications

Выполнила Иванова Виктория
Гр. 146 ФАУ

Microsoft Visual Basic — средство разработки программного обеспечения, разрабатываемое корпорацией *Microsoft* и включающее язык программирования и среду разработки. Язык *Visual Basic* унаследовал дух, стиль и отчасти синтаксис своего предка — языка Бейсик.

В то же время *Visual Basic* сочетает в себе процедуры и элементы объектно-ориентированных и компонентно-ориентированных языков программирования.

Visual Basic считается хорошим средством быстрой разработки прототипов программы, для разработки приложений баз данных и вообще для компонентного способа создания программ, работающих под управлением операционных систем семейства *Microsoft Windows*.

Эволюция Visual Basic

- **май 1991** — выпущен *Visual Basic 1.0* для *Microsoft Windows*. За основу языка был взят синтаксис *QBasic*, а новшеством, принесшим затем языку огромную популярность, явился принцип связи языка и графического интерфейса. Этот принцип был разработан Аланом Купером (Alan Cooper) и реализован в прототипе Tripod (также известном как Ruby). Первый *Visual Basic* был интерпретатором.

- **сентябрь 1992** — выпущен *Visual Basic 1.0* под DOS. Он не был полностью совместим с Windows-версией VB, поскольку по сути являлся следующей версией QuickBASIC и работал в текстовом режиме экрана. Интересно отметить, что в составе была специальная утилита, которая конвертировала проекты данной версии в проекты *Visual Basic 1.0*. Она делала код максимально совместимым с Windows-версией, конвертировала формы (хотя результат получался неудовлетворительным из-за различий текстового и графического режима), а также давала в коде комментарии с подсказками по дальнейшему улучшению.

- **ноябрь 1992** — выпущен *Visual Basic 2.0*. Среда разработки стала проще в использовании и работала быстрее.
- **летом 1993** — вышел в свет *Visual Basic 3.0* в версиях Standard и Professional. Ко всему прочему, в состав поставки добавился движок для работы с базами данных Access.
- **август 1995** — *Visual Basic 4.0* — версия, которая могла создавать как 32-х так и 16-разрядные Windows-программы. Кроме того, появилась возможность писать на VB классы, а также *Visual Basic* наконец-то стал полноценным компилятором, что значительно увеличило скорость выполнения программ.

- **февраль 1997** — *Visual Basic 5.0* — начиная с этой версии, стало возможно, наряду с обычными приложениями, разрабатывать COM-компоненты. Скомпилировав такой компонент в OCX-файл и поставляя его, можно было предоставить свой объект управления не только конечному пользователю, но и другим разработчикам, после чего они могли интегрировать этот объект в свои приложения.

- **В середине 1998** — вышла *Visual Basic 6.0.* После этого Microsoft резко изменила политику в отношении языков семейства Basic. Вместо развития Visual Basic, был создан абсолютно новый язык Visual Basic .NET, первая версия которого появилась в 2001. Это принципиально новый язык, имеющий, помимо синтаксиса, очень мало схожего с VB 6.0; и отличаясь от него так же сильно, как в своё время VB отличался от QBASIC. VB.NET сразу же занял место VB в Visual Studio, и на этом развитие классического Visual Basic остановилось.

- **В 2005 году** вышла новая версия *Visual Basic*, в комплекте *Visual Studio*. Порадовала она новым интерфейсом и возможностями. Язык основан на *Visual Basic.NET*. Она не была классической версией VB.
- **В конце 2007** Microsoft выпустила новую версию *Visual Basic — Visual Basic 2008,* которая также была основана на *Visual Basic.NET*.

Visual Basic for Applications (VBA, Visual Basic для приложений) — немного упрощённая реализация языка программирования *Visual Basic,* встроенная в линейку продуктов *Microsoft Office* (включая версии для Mac OS), а также во многие другие программные пакеты, такие как AutoCAD, SolidWorks, CorelDRAW, WordPerfect и ESRI ArcGIS. VBA покрывает и расширяет функциональность ранее использовавшихся специализированных макро-языков, таких как WordBasic.

VBA является интерпретируемым языком. Как и следует из его названия, VBA близок к *Visual Basic*. VBA, будучи языком, построенным на COM, позволяет использовать все доступные в операционной системе COM объекты и компоненты ActiveX. По сути, возможно создание приложения на основе Microsoft Word VBA, использующего только средства Corel Draw.

В будущем Microsoft планирует заменить VBA на *Visual Studio Tools for Applications (VSTA)* — инструментарий расширения функциональности приложений, основанный на Microsoft .NET.

- Приложение, создаваемое в среде Visual Basic, называется проектом.

- Программный проект — это совокупность частей, составляющих будущее WINDOWS-приложение.

- Любой проект должен обязательно состоять из экранных форм (хотя бы одной) и программных модулей (хотя бы одного).

Создание любого приложения состоит из следующих этапов:

- Постановка задачи;
- Разработка интерфейса;
- Программирование;
- Отладка программы;
- Сохранение проекта и при желании — компиляция (превращение проекта в исполняемое приложение, способное работать самостоятельно за пределами среды проектирования).

Процедура – изолированная часть программы, предназначенная для решения локальной задачи.

Общий вид процедуры обработки события:

Sub Объект_Событие(параметры)

Действия (команды языка VB)

End Sub

Правила работы в *Visual Basic:*

1.Текстовая информация должна быть заключена в двойные кавычки.

2.При написании команд и имен регистр символов не учитывается (DIM и dim – одно и то же).

3.Имена переменных и подпрограмм должны начинаться с буквы и не содержать пробелов.

4.Количество пробелов и пустых строк между командами может быть любым.

5.Обычно каждая команда записывается в отдельной строке.

6.Комментарии – это пояснения в тексте программы. Комментарий должен начинаться с символа ‘ (одиночная кавычка), тогда *Visual Basic* не воспринимает этот текст как программу и просто игнорирует его. Комментарий действует только до конца строки.

Обычно программы на алгоритмических языках программирования выполняются от начала до конца согласно алгоритму. Приложение на *Visual Basic* работает совершенно по-другому: выполнение приложения не идет предопределенным путем, а зависит от сложившейся ситуации. Программа является набором независимых частей кода, активизирующихся в ответ на события, происходящие при работе пользователя с приложением. Поэтому вместо создания программы, отражающей мысли программиста, получается приложение, управляемое пользователем.