


# Visual BASIC

## **Закон программирования гласит:**


1. Ни одна, даже самая простая программа, не работает сразу после написания.
2. Легче написать свою собственную программу, чем разбирать и исправлять чужую.


## *ОБЩАЯ ХАРАКТЕРИСТИКА ЯЗЫКА.*

---

 *BASIC* представляет собой простой язык программирования, разработанный в 1964 году для использования новичками.

- *Название языка BASIC является аббревиатурой от **B**eginners **A**ll-purpose **S**ymbolic **I**nstruction **C**ode — многоцелевой язык символьческих инструкций для начинающих.*

 *BASIC* был разработан как простейший язык для непосредственного общения человека с вычислительной машиной.

- *Первоначально работа велась в интерактивном режиме *

## *ОБЩАЯ ХАРАКТЕРИСТИКА ЯЗЫКА.*

---



*Согласно концепциям, заложенным в Basic, этот язык в смысле строгости и стройности является антиподом языка Pascal.*

- *В частности, в нем широко распространены различные правила умолчания, что считается плохим тоном в большинстве языков программирования подобного типа.*




*Существует множество диалектов этого языка, мало совместимых между собой.*

- *Basic активно поглощает многие концепции и новинки из других языков.*

## *ИСТОРИЯ РАЗВИТИЯ ЯЗЫКА.*

---

 *Первая версия языка (VB 1.0) была выпущена фирмой Microsoft в 1992 г.*

- *VB 1.0 - очень простой язык программирования для Windows 3.1.*

 *Затем были выпущены VB 3.0, VB 4.0, VB 5.0 и, наконец, VB 6.0.*

- *Последние две версии мало чем отличаются, в шестой версии улучшено ядро и добавлены несколько новых функций.*



*Существует ещё язык  
макросов для приложений  
Microsoft Office.*

*Он называется VBA  
(Visual Basic for Application).*

## *ОБЩАЯ ХАРАКТЕРИСТИКА ЯЗЫКА.*



*Приложения написанные на Visual Basic отличаются от обычных приложений тем, что требуют для своей работы библиотеку **msvbvmX0.dll**, которая должна присутствовать в каталоге **Windows/System**.*

- *В роли X выступает версия компилятора VB. Для **VB5** - **msvbvm50.dll**, для **VB6** - **msvbvm60.dll**.*
- *Эти библиотеки идут в комплекте с **WinME** (обе) и **Win98** (только **msvbvm50.dll**), и естественно с более новыми версиями этих ОС..*



*Вы можете зашить библиотеку прямо в **EXE** файл.*



*Права для этого придется использовать средства из*

*Некоторые программисты считают, что приложения VB - это не полноценные программы, а лишь псевдокод, который при запуске выполняется интерпретатором.*

- Это не совсем так.
- Если вы компилируете программу в **Native Code**, то полученный **EXE** - полноценное приложение **Win32**, которое просто использует функции **msvbvm** библиотеки.
- А вот **P-Code** является псевдокодом. Вид компиляции вы можете указать в меню **Project->ProjectX Properties...**

- Visual Basic доступен в трех редакциях, каждая из которых обеспечивает определенный набор инструментов разработки.

- **Standard Edition**

- Professional Edition
- Enterprise Edition



## Standard Edition

- позволяет создавать достаточно мощные приложения для Windows.
- включает все встроенные элементы управления Visual Basic, включая связанные (data-bound) элементы управления.

## Professional Edition

- обеспечивает полный набор инструментальных средств для разработки профессиональных решений, предназначенных для тиражирования.
- включает все возможности Standard Edition плюс дополнительные элементы управления ActiveX, включая элементы управления для Internet и генератор отчетов Crystal Reports для Visual Basic.

## Enterprise Edition

- позволяет создавать распределенные приложения силами группы разработчиков.
- обеспечивает все возможности редакции Professional и включает дополнительные функции, такие, как **Automation Manager**, **Component Manager**, инструментальные средства управления базами данных и **Microsoft Visual SourceSafe** — проект-ориентированная система управления версиями продуктов.

## Возможности Visual Basic ничем не ограничены

- Вы можете расширять возможности VB посредством использования дополнительных функций.
- Visual Basic позволяет использовать библиотеки

На VB можно написать любую программу, от обслуживающих рутинные операции ввода данных, до сложных информационных и коммуникационных систем.

- В США 60% программных продуктов написаны на VB.

Есть, конечно, очень небольшие ограничения

- на VB нельзя написать:
  - полноценную DLL библиотеку;
  - драйвер устройства VXD.
- также в VB нельзя использовать ассемблер.

В Visual Basic, как и во многих других языках, предназначенных для написания приложений под Windows, используется **событийно-управляемая модель** программирования.

- В "старых" версиях бейсика, таких как **QBasic**, использовалась плоская структура написания программы.
- Каждая программа начиналась и заканчивалась в определённых местах.
- Вся программа выполнялась последовательно, и иногда, возможно, вызывались пользовательские процедуры и функции.
- *То есть если программу "запускали", то она сразу начинала выполняться, и, дойдя до конца, завершалась.*

В Visual Basic программа строится совсем иначе.

- Дело в том, что ОС Windows имеет **GUI** (***Graphical User Interface***), т.е. графический интерфейс пользователя, в котором используются стандартные элементы управления, такие, как окна (они же формы), кнопки, списки, поля, для ввода текста и т.п.
- В любом современном языке высокого уровня программа строится на основе этих элементов.

## Разработка приложения на VB состоит из следующих этапов:

- 1. Продумывание программы** (подумать, что программа должна делать, решить перед собой задачи, реализовать их мысленно, продумать структуру данных, и т.д.).
- 2. Проектирование интерфейса**, т.е. помещение на форму нужных управляющих элементов, кнопок, списков и т.п. (этот этап называется составлением скелета программы).
- 3. Написание программного кода**, связывающего помещённые на форму управляющие элементы, т.е. "наращивание плоти на скелет".
- 4. Отладка программы** (этот этап часто занимает больше времени, чем предыдущие).
- 5. Окончательная компиляция** и, если это необходимо, **создание дистрибутива** (т.е. установочного файла *setup.exe*).

Этапы разработки приложения всегда необходимо держать в памяти

- т.к. они используются при написании любой программы

Порядок этапов тоже важен.

- Например, нельзя сначала написать код программы, а только потом разрабатывать интерфейс пользователя.
- Есть, конечно, редкие исключения, например, когда программа вообще не имеет интерфейса, или когда вы пишете процедуру, независимую от интерфейса программы.

**Этапы разработки приложения.**



В Visual Basic любой проект состоит из следующих файлов:

Файл	Расширение	Примечание
формы	<b>frm</b>	обычный текстовый файл, в котором записан весь код, помещённый в форму, а также свойства всех помещённых на форму элементов управления и самой формы тоже.
формы, содержащий бинарную информацию	<b>frx</b>	В качестве бинарной информации может выступать например картинка в PictureBox.
проекта	<b>vbp</b>	содержит информацию о проекте
информация о рабочей области проекта	<b>vbw</b>	workspace

Это необходимый минимум.

*Хотя, бывают и исключения, например, когда в проекте не используются формы, тогда вместо **frm** файла, будет **bas** файл.*

**Структура проекта на Visual Basic.**

Дополнительные файлы, которые могут быть подключены к проекту :

Файл	Расширение	Примечание
модуля	<b>bas</b>	текстовый файл.
модуля классов	<b>cls</b>	текстовый файл.
дополнительного элемента управления	<b>ctl</b>	текстовый файл.
ресурсов	<b>res</b>	
другие файлы	<b>ocx, tlb, ...</b>	

**Структура проекта на Visual Basic.**

На начальном этапе,  
достаточно запомнить  
два файла:

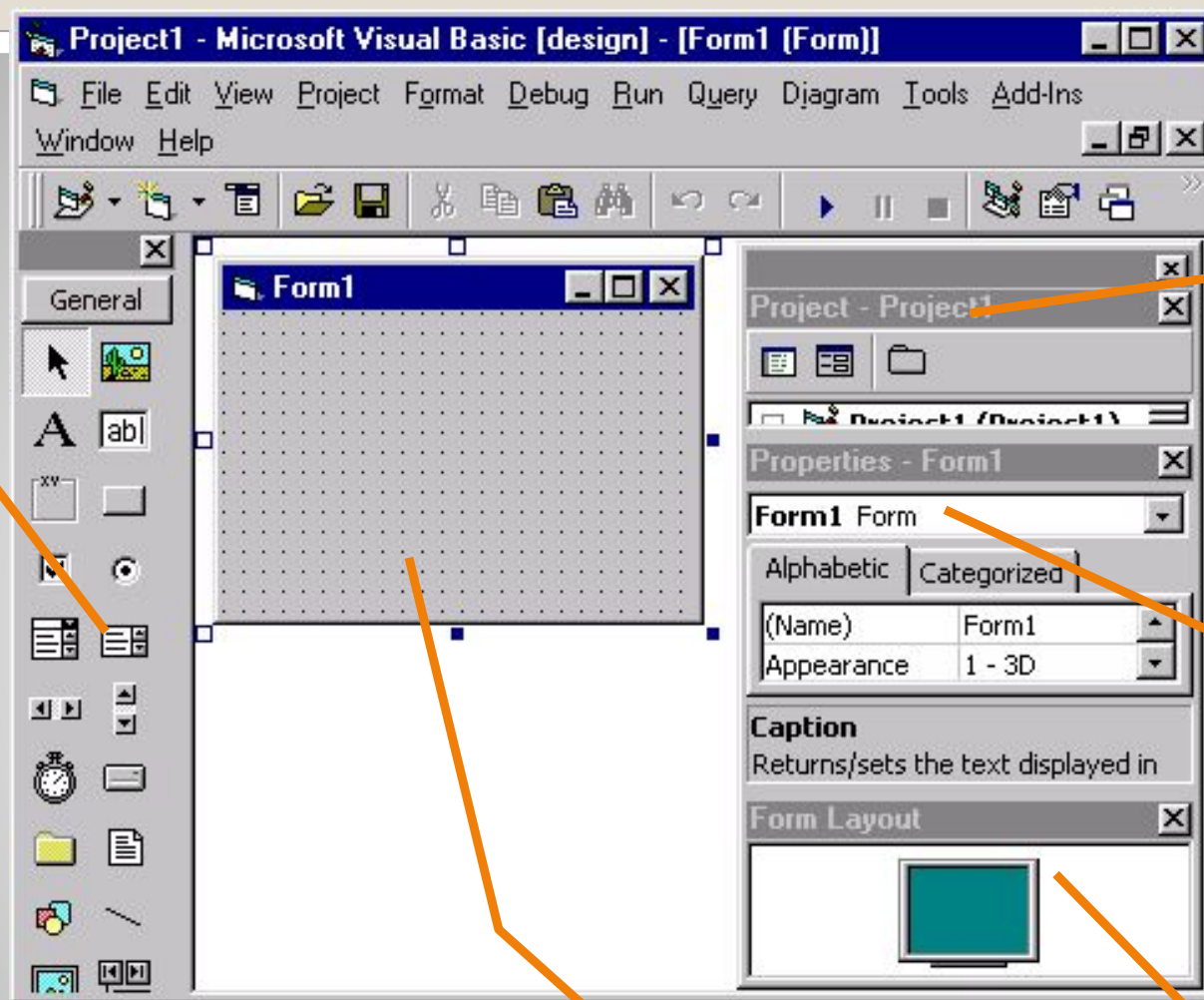
- 1. frm-файл**, в котором хранятся код формы и свойства всех помещённых на данную форму элементов управления.
- 2. bas-файл** — модуль, в котором могут быть объявлены глобальные переменные, константы, функции и т.д. (короче говоря, только код, без элементов управления).

Для того, чтобы понять как проектировать интерфейс, разберём среду разработки Visual Basic:

- 1. Запускаем VB** — Перед вами появится окошко, в котором вас попросят указать тип проекта.
- 2. Указываем тип "Standart EXE»**
- 3. Нажмаем ОК.**

*панель  
элементов  
управления*

*окно  
проводника  
проекта*



*окно свойств  
текущего  
элемента  
управления  
или формы*

*окна, содержащие формы,  
модули и другие элементы  
проекта.*

*окно  
размещения  
формы на  
экране  
монитора.*

**Окно среды программирования Visual Basic.**

Все объекты в VB имеют  
**свойства и методы.**

- **Свойства** — значения, которые устанавливаются для определения вида и поведения объекта.
- **Методы** — программные процедуры, обеспечивающие выполнение объектом некоторых predetermined действий.

Главная задача разработчика программы - установить свойства объекта и вызвать методы объекта, чтобы побудить объект выполнить требуемые функции.

Чтобы изменить свойства элемента управления необходимо выделить его и изменить нужное свойство в окне Properties.

- Некоторые свойства можно отредактировать только в период выполнения программы (RunTime).

Многие объекты имеют одинаковые свойства.

- Например, свойство **Caption**.
  - У формы (**Form**) **Caption** - это заголовок окна,
  - У элемента метки (**Label**) - это текст внутри метки.