

INTRODUCTION

VISUAL BASIC is a high-level programming language which evolved from the earlier DOS version called BASIC (Beginners' All- purpose Symbolic Instruction Code). However, people prefer to use Microsoft Visual Basic today, as it is a well-developed programming language and supporting resources are available everywhere.

Visual Basic is easy to learn Programming language. With Visual Basic you can develop Windows based applications and games. Visual Basic is much easier to learn than other language (like Visual C++), and yet it's powerful programming language.

Now, there are many versions of VB exist in the market, the most popular one and still widely used by many VB programmers is none other than Visual Basic 6 . We also have VB.net, Visual Basic 2005, Visual Basic 2008 , Visual Basic 2010, Visual Basic 2012 and Visual Basic 2013 . VB2008, VB2010, VB2012 and VB2013 are fully object-oriented programming (OOP) languages.

EVOLUTION OR HISTORY OF VISUAL BASIC

BASIC (Beginners All Purpose Symbolic Instruction Code) was developed in 1960's by **Profs Kemeny & Kurtz**. 1970's Bill gates implemented BASIC in several PCs.

Alan Cooper is considered the father of Visual Basic. In 1987, the then Director of Applications Software for Coactive Computing Corporation wrote a program called **Ruby (Tripod)** that delivered visual programming to the average programmer/user. **Alan Cooper** developed VB and sold to Microsoft in **1988**.

The Visual Basic (VB) system is a **fourth-generation programming** system which produces much of the code itself as the programmer designs the interface for his or her application. Microsoft surveys in the late 1990's showed that roughly two-thirds of all business applications programming on PCs was being done in Visual Basic.

Version	Year	New Features
VB 1.0	1991	The interface was barely graphical, using extended ASCII characters to simulate the appearance of a GUI.
VB 2.0	1992	The programming environment was easier to use, and its speed was improved.
VB 3.0	1993	VB3 included a database engine that could read and write Access databases.
VB 4.0	1995	32bit and It also introduced the ability to write classes in Visual Basic.
VB 5.0	1996	The ability to create custom user controls, as well as the ability to compile to native Windows executable code, speeding up runtime code execution.
VB 6.0	1998	Improved in many areas including the ability to create web-based applications using Internet Explorer. Visual Basic 6 is no longer supported.

INTEGRATED DEVELOPMENT ENVIROMENT

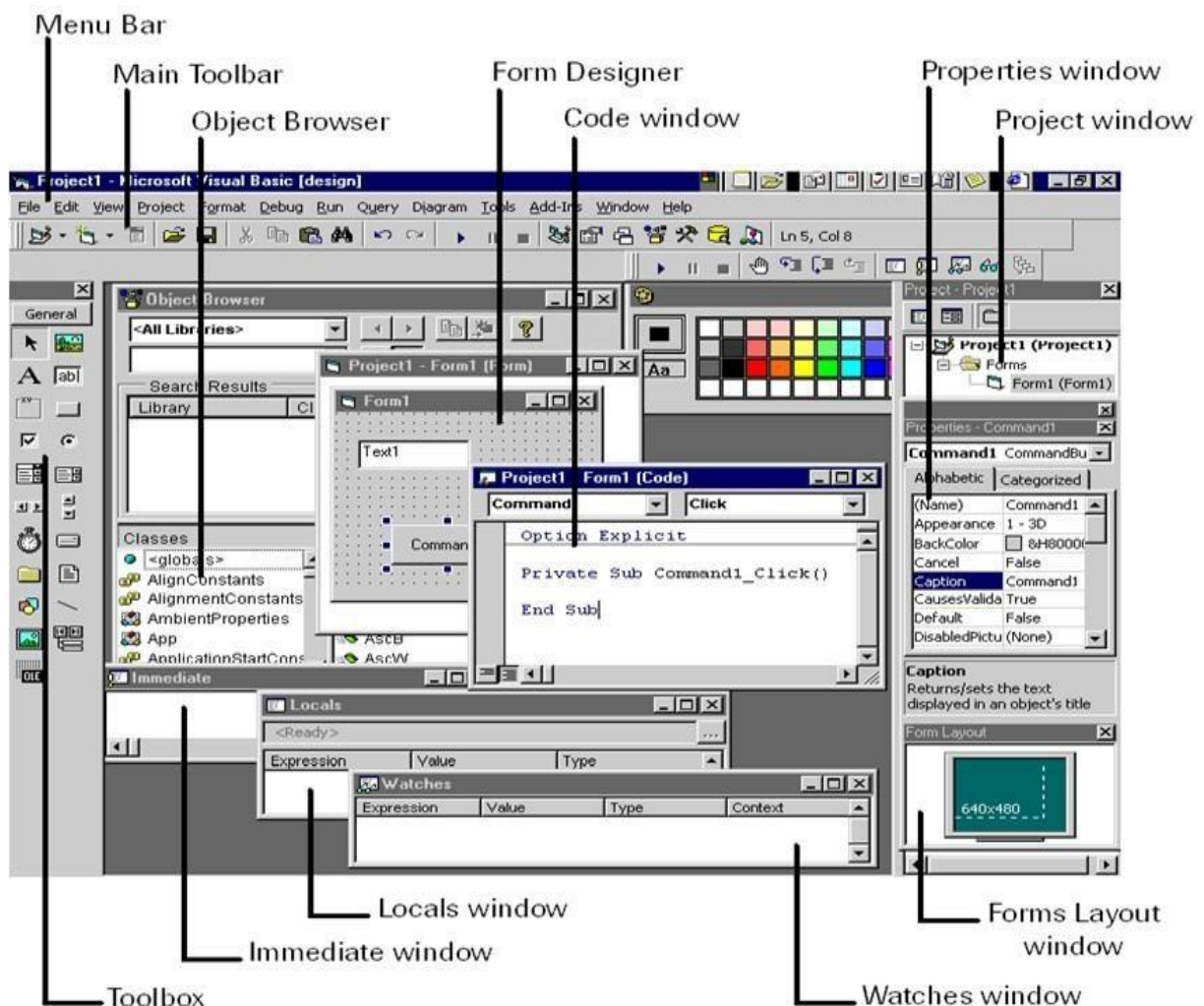
An **Integrated Development Environment (IDE)** is a software application that provides comprehensive facilities to computer programmers for software development. An IDE normally consists of a source code editor, build automation tools and a debugger.

WHY VB IS CALLED IDE?

One of the most significant changes in Visual Basic 6.0 is the Integrated Development Environment (IDE). IDE is a term commonly used in the programming World to describe the interface and environment that we use to create our applications. It is called **integrated** because we can access virtually all of the development tools that we need from one screen called an interface. The IDE is also commonly referred to as the design environment, or the program.

Integrated Development Environment (IDE) consists of inbuilt compiler, debugger, editors, and automation tools for easy development of code. Visual Basic.net 2006 IDE can be accessed by opening a new project. IDE was first introduced with version 5.0 and Integrated Development Environment of Visual Studio.net 2008 had undergone minor design changes. VB IDE consists of Solution Explorer, Toolbox, Form, Properties Window, and Menu Bar. In Visual Studio windows related to a project are combined together and placed at certain locations on the screen. This type of IDE is known as Multiple Document Interface or MDI. It also having the great feature called as **drag and drop**. We can drag & the drop the controls without writing single line of coding.

Elements of Integrated Development Environmental (IDE)



The Visual Basic IDE is made up of a number of components

- Menu Bar
- Tool Bar
- Project Explorer
- Properties window
- Form Layout Window
- Toolbox
- Form Designer
- Object Browser

In previous versions of Visual Basic, the IDE was designed as a Single Document Interface (SDI). In a Single Document Interface, each window is a free-floating window that is contained within a main window and can move anywhere on the screen as long as Visual Basic is the current application. But, in Visual Basic 6.0, the IDE is in a Multiple Document Interface (MDI) format. In this format, the windows associated with the project will stay within a single container known as the parent. Code and form-based windows will stay within the main container form.

ADVANTAGES OF VB

1. The structure of the **Basic programming language** is very simple, particularly Visual Programming with VB

- as to the executable code.
2. VB is not only a language but primarily an integrated, interactive development environment (“**IDE**”).
 3. The VB-IDE has been highly optimized to support rapid application development (“RAD”). It is particularly easy to **Develop Graphical User Interfaces** and to connect them to handler functions provided by the application.
 4. The graphical user interface of the **VB-IDE** provides intuitively appealing views for the management of the program structure in the large and the various types of entities (classes, modules, procedures, forms,).
 5. It is an **Event Driven Programming** which provides complete control to the end user.
 6. VB is a first **Programmer friendly** language in the world.
 7. VB provides a comprehensive **interactive and context-sensitive** online help system.
 8. When editing program texts the “**IntelliSense**” technology informs you in a little popup window about the types of constructs that may be entered at the current cursor location.
 9. Visual Basic 6.0 features provide graphical, integrated data access to any **ODBC or OLE DB** data source, and additional database-design tools for **Oracle and Microsoft SQL Server-based databases**.
 10. New Web development features bring the easy-to-use, component-based programming model of Visual Basic to the creation of **HTML- and Dynamic HTML (DHTML)-based applications**
 11. VB is a component integration language which is attuned to Microsoft’s **Component Object Model (“COM”)**.
 12. COM components can be written in different languages and then integrated using VB.
 13. Interfaces of COM components can be easily called remotely via Distributed COM (“DCOM”), which makes it easy to construct distributed applications.
 14. COM components can be embedded in / linked to your application’s user interface and also in/to stored documents (**Object Linking and Embedding** “OLE”, “Compound Documents”).
 15. There is a wealth of readily available COM components for many different purposes.
 16. Visual Basic is built around the .NET environment used by all Microsoft Visual languages, so there is very little that can’t be done in Visual Basic that can be done in other languages (such as C#).

DISADVANTAGES OF VB

1. Visual basic is a proprietary programming language written by Microsoft, so programs written in Visual basic cannot, easily, be transferred to other operating systems. It's a platform dependent it only runs on MS Windows operating system.
2. There are some, fairly minor disadvantages compared with C. C has better declaration of arrays – it's possible to initialize an array of structures in C at declaration time; this is impossible in VB.

EVENT DRIVEN PROGRAMMING

Event-driven programming is a programming paradigm in which the flow of program execution is determined by *events* - for example a user action such as a mouse click, key press, or a message from the operating system or another program is known as the Event Driven Programming. VB programming is also based on Events.

An event-driven application is designed to detect events as they occur, and then deal with them using an appropriate *event-handling procedure*.

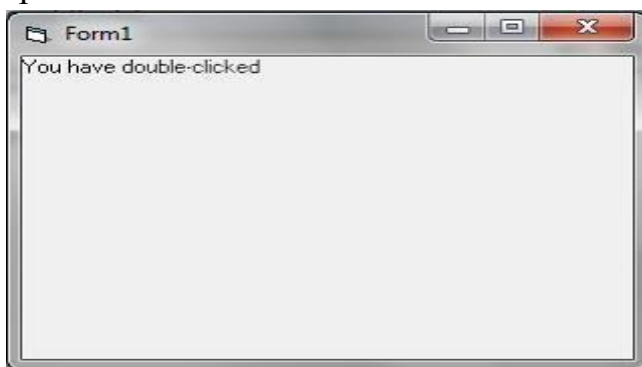
When you fire an event, the code in the event procedure is executed, and then visual basic performs its operations as per the instructions written in the event procedure code. For example, in the first sample program, when you click the 'Print' button, the click event is fired, and then the code in the click event procedure gets executed. The code tells Visual Basic to print a text on the form. So as a result, you see a text printed on the form.

Example:

Write the following code in the DblClick event procedure of the form.

```
Private Sub Form_DblClick()  
    Print "You have double-clicked"  
End Sub
```

Output:



When you double-click on the form, the DblClick event procedure of the Form object is invoked, and then the code in the DblClick event procedure is executed. Thus, the code instructs Visual Basic to print a text on the form.

CHARACTERISTICS AND FEATURES OF VISUAL BASIC

Visual Basic (VB) is a unique computer language---at least it was when it first came out. Now there are many imitators. VB allows you to quickly and easily develop a bank of visual controls with sliders, switches and meters or a complex form for a user to fill out. It uses the BASIC language which is known to most computer programmers, and which can be learned quickly if it is not already known.

IDE

Integrated Development Environment (IDE) consists of inbuilt compiler, debugger, editors, and automation tools for easy development of code. Visual Basic.net 2006 IDE can be accessed by opening a new project. IDE was first introduced with version 5.0 and Integrated Development Environment of Visual Studio.net 2008 had undergone minor design changes. VB IDE consists of Solution Explorer, Toolbox, Form, Properties Window, and Menu Bar. In Visual Studio windows related to a project are combined together and placed at certain locations on the screen. This type of IDE is known as Multiple Document Interface or MDI. It also having the great feature called as **drag and drop**. We can drag & the drop the controls without writing single line of coding.

GUI Interface or User Interface

VB is a Graphical User Interface (GUI) language. This means that a VB program will always show something on the screen that the user can interact with (usually via mouse and keyboard) to get a job done. The first step in building the VB program is to get the GUI items on the screen. This is done via pull-down menus that list the available graphical objects. Every system is slightly different (Mac differs from Windows and VB4 Differs from VB6) but, generally speaking, left-clicking on an object allows you to describe attributes like size and position. Right clicking allows you to write code. For example, if the GUI item is a switch, left-clicking would allow the programmer to say how big the switch was, how it was labeled and where on the screen it is positioned. Right-clicking on the switch would bring up a window that allows the programmer to write the code that describes what happens when the user clicks the switch.

Object Oriented

Object Oriented Programming (OOP) is a concept where the programmer thinks of the program in "objects" (however abstract the objects may be) that interact with each other. In OOP, all the code associated with that object is in one place. Once again, VB forces this good programming practice. The GUI items are the objects and all the code associated with the object are just a click away. This natural way of enforcing good programming practices--- plus the ease of programming in BASIC---is exactly why VB has found so many devoted fans.

Event Driven Programming

Event-driven programming is a programming paradigm in which the flow of program execution is determined by *events* - for example a user action such as a mouse click, key press, or a message from the operating system or another program is known as the Event Driven Programming. VB programming is also based on Events.

An event-driven application is designed to detect events as they occur, and then deal with them using an appropriate *event-handling procedure*.

Modularization

It is considered good programming practice to modularize your programs. Small modules where it is clearly indicated what comes into the module and what goes out makes a program easy to understand.

Debugging

Visual Basic offers two different options for code debugging:- Debugging Managed Code Runtime Debugger The Debugging Managed Code individually debugs C and C++ applications and Visual Basic Windows applications. The Runtime Debugger helps to find and fix bugs in programs at runtime.

Data Access

By using data access features, we can create databases, scalable server-side components for most databases, including Microsoft SQL Server and other enterprise-level database.

Macros IDE

The Macros integrated development environment is similar in design and function to the Visual Studio IDE. The Macros IDE includes a code editor, tool windows, the properties windows and editors.