

Chapter 1 Introduction to VB 2010

❖ Introduction

In computing, a Visual Programming Language (VPL) is any programming language that lets users create programs by manipulating program elements *graphically* rather than by specifying them *textually*. A VPL are based on the idea of boxes and other objects, where boxes or other screen objects are treated as entities which connected in some ways by relations.

VPLs may be further classified, according to the type and extent of visual expression used, into *icon-based languages*, *form-based languages*, and *diagram languages*. Visual programming environments provide graphical or iconic elements which can be manipulated by users in an interactive way according to some specific spatial grammar for program construction. An important benefit of learning Visual Basic and the Visual Studio *Integrated Development Environment (IDE)* is that you can use many of the same tools to write programs for Microsoft Visual C++ 2010, Microsoft Visual Basic 2010, Microsoft Visual Web Developer 2010, and other popular products.

Visual programming languages are widely used for the rapid development of graphical applications. This subject will introduce students to the fundamental principles of the term **Event-Driven Programming** (*Events in this context include clicking a button, resizing a window, or changing an entry in a text box. The code that you write responds to these events. The program reacts to your movements and takes the necessary actions to complete your desired tasks*) and to programming using a **Visual Environment** through the use of the Visual Basic programming language. An additional aim of this subject is to give students an understanding of the main ideas of Human Computer Interaction (**HCI**).

❖ Visual Basic vs. VB .NET

Visual Basic (**VB**) was initially introduced in 1991 as the first programming language that directly supported programmable *Graphic User Interface (GUI)* using language supplied object. VB language, greatly make the creation of windows applications easy especially, by facilitating the use of re-usable components.

Visual Basic.NET (**VB.Net**) a programming language based on VB 6.0 working on the .NET framework of the Microsoft Corporation. This framework represents a platform for cross-language development (C#, VB. NET, C++, F#) and includes a large standard library called the Base Class Library (**BCL**).

Visual Studio intended mainly for *Windows Applications and Web Applications*. We will use Visual Studio 2012 to create all of our programs.

Visual Basic is an Object Oriented Language (**OOL**) that let you program with many different program groups called *Modules* and *Classes*. Visual Basic consists of two fundamental parts: a **visual part** *consists of set of objects* and **language part** *which consists of high level procedural programming language*. These two parts are used together to create **Application** *which is simply a visual basic program that can run under the windows operating system*.

❖ **Prerequisites**

All you need one version of Visual Studio 2012 Professional, Visual Studio 2012 Premium, or Visual Studio 2012 Ultimate to execute the exercises, if you don't have, you can complete most of the exercises by downloading Visual Basic 2012 Express from the Web for free. This will give you an opportunity to learn Visual Basic programming and see for yourself if you want to upgrade to a full release of the Visual Studio software.

To download Visual Basic 2012 Express, complete the following steps:

1. Open a Web browser and go to <http://www.microsoft.com/express>.
2. Follow the instructions on the screen to download Visual Basic 2012 Express. On the Express Web site, you will also see an Express product feature chart that compares the Express product to the full versions of Visual Studio. Although there are some key differences between the full versions and Visual Basic 2012 Express, many of these differences have no effect on how you learn the essential techniques and features of Visual Basic programming. After you experiment with the Express product, you can decide whether you want to upgrade to one of the full versions of Visual Studio or not.

❖ **Microsoft Visual Studio & Visual Basic (.NET Version)**

Microsoft's Visual Studio (also called Visual Studio.NET) includes several different programming languages:

- **Visual Basic**
- **Visual C#**
- **Visual C++**
- **Visual F#**
- **Jscript**
- **Web Development (called ASP.NET)**
- It also includes the **.NET 4 Framework** upon which these languages operate.

All of these languages **compile**. This means they are translated from human readable-form to machine readable-form to the same Microsoft Intermediate Language (MSIL),

then, MSIL run within the Common Language Runtime (CLR) – a component of the .NET Framework.

VB is also termed an **event-driven** programming language because you will write program code that responds to events that are controlled by the system user. Example events include:

- Clicking a button or menu.
- Opening or Closing a form.
- Moving the mouse over the top of an object such as a text box.
- Moving from one text box to another.

In order to work with VB, you need to understand "object" terminology as defined below.

- **Object** A thing – like a noun in English like forms and controls you place on forms such as buttons, text boxes, and icons.
- **Property** Objects have properties – like adjectives in English. Properties describe object behaviors. For examples Text, Name, BackColor, Font, and Size. Refer to a property by the notation `ObjectName.PropertyName` (use the .dot notation) – like `TotalTextBox.Text` or `AccountLabel.ForeColor`. Each property has value.
- **Method** Like a verb in English – these are the actions that objects exhibit. For examples methods to Show or Hide forms and methods to Print or Close forms. Refer to a method with the notation `ObjectName.MethodName` – example `Me.Close` will close the current form.
- **Event** Events are actions usually triggered by the system user such as clicking a button; however, events can also be triggered by the actions of objects. For example, closing a form can trigger an event.
- **Class** This is a really abstract term – it is a sort of template for an object. For example, all forms belong to the Form class of object. All buttons belong to the Button class of object. Classes include definitions for object properties, methods, and associated events. Each class is assigned an identifying namespace within the .NET Framework Class Library. Each new object you create is defined based on its class – the new object is called a class instance.