LONDON CAPITAL COMPUTER COLLEGE

Diploma in Programming (601) – Visual Basic .Net

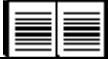|	
Prerequisites: Basic programming skills or basic knowledge of computer use.	**Corequisites:** A pass or higher in Diploma in System Analysis & Design or equivalence.
Aim: This course introduces the candidates to the development of Visual Basic applications using the .Net framework. The course provides basic understanding of how to use Visual Studio .Net and write simple programs using VB .Net programming language. Candidates are encouraged to grasp the functionality and syntax of VB .Net and develop desktop and web-based applications.	
Required Materials: Student study materials	**Supplementary Materials:** Recommended textbooks and lecture notes.
Special Requirements: This is a hands-on course, hence practical use of computers is essential. Requires intensive lab work outside of class time.	

Intended Learning Outcomes:	Assessment Criteria:	
1. Compare Visual Basic .Net to other high-level languages. Define structured programming. Discuss Microsoft .Net.	1.1	Navigate Visual Studio .Net's **start page**
	1.2	Describe how to create a Visual Basic .Net solution
	1.3	Identify how to use the IDE's menus and toolbars
	1.4	Identify how to manipulate windows in the Visual Studio .Net IDE
	1.5	Describe how to se the auto hide feature
	1.6	Describe how to use the visual studio .Net IDE's help features
	1.7	Describe how to close a Visual Basic .Net solution.
2. Outline the overview of Visual Studio .Net. Describe the menu bar, toolbar, how to save and close Visual Studio .Net	2.1	Describe how to set the text in the form's title bar
	2.2	Define how to change the form's background color
	2.3	Describe how to place a label control on the form
	2.4	Demonstrate how to display text in a label control
	2.5	Demonstrate how to place a picturebox control on the form
	2.6	Identify how to display an image in a picturebox control
	2.7	Demonstrate how to execute an application.
3. Discuss how to construct a simple application. Identify how objects are used.	3.1	Analyse how to visually program, using GUI design guidelines
	3.2	Identify how to rename a form and set the form as the startup object
	3.3	Demonstrate how to add labels, textboxes, and a button to the form
	3.4	Describe how to use the textalign and borderstyle properties for labels.

4. Describe how to add Labels, Textboxes and Buttons to a form.	4.1	Describe how to add an event handler for a button control	
	4.2	Identify how to insert code into an event handler	
	4.3	Outline how to access a property's value by using Visual Basic .Net code	
	4.4	Demonstrate how to use the assignment and multiplication operators.	
5. Set options for Visual Studio .Net environment. Describe event handler, debugger and syntax errors.	5.1	Describe how to create variables	
	5.2	Outline how to handle the textchanged event	
	5.3	Illustrate how to apply basic memory concepts using variables	
	5.4	Describe the precedence rules of arithmetic operators	
	5.5	Demonstrate set breakpoints to debug applications	
6. Define variables, memory concepts and arithmetic operations. Discuss algorithms, pseudocode, program control structures, assignment operators and formatting text.	6.1	Demonstrate basic problem-solving techniques	
	6.2	Describe control structures	
	6.3	Define and create pseudocode	
	6.4	Demonstrate how to use the *if…then* and *if…then…else* selection statements to choose among alternative actions	
	6.5	Identify how to use the assignment operators	
	6.5	Identify how to use the debugger's **watch** window.	
7. Illustrate how to use CheckBoxes and message dialogs. Describe logical operators.	7.1	Demonstrate how to use checkboxes to allow users to select options	
	7.2	Demonstrate how to use dialogs to display messages	
	7.3	Describe how to use logical operators to form more powerful conditions.	
8. Define Do While …. Loop and Do Until … Loop repetition statements.	8.1	Describe how to use the do **while…loop** and **do until…loop** repetition statements to execute statements in a program repeatedly	
	8.2	Define how to use counter-controlled repetition	
	8.3	Demonstrate how to display information in listboxes	
	8.4	Demonstrate how to use the **do…loop while** statement	
	8.5	Demonstrate how to use the **do…loop until** statement	
	8.6	Describe counter-controlled repetition	
	8.7	Analyse how to transfer the focus to a control	
	8.8	Demonstrate how to enable and disable buttons.	
9. Understand The For…Next statement. Describe the counter-controlled repetition.	9.1	Demonstrate how to execute statements repeatedly with the **for…next** repetition statement	
	9.2	Describe how to obtain user input with	

			the numeric updown control
		9.3	Describe how to display information, using a multiline textbox.
10. Describe the multiple-selection statement.		10.1	Describe how to use the select case multiple-selection statement
		10.2	Demonstrate how to use the *is* keyword
		10.3	Analyse how to use case
		10.4	Illustrate how to use the textbox property passwordchar statement
		10.5	Demonstrate how to display a date and time
11. Describe classes, procedures, function procedures and sub procedures.		11.1	Demonstrate how to construct applications modularly from pieces called procedures
		11.2	Analyse how to work with "built-in" procedures Distinguish between function procedures and sub procedures, and determine when each should be used;
		11.4	Describe how to create your own function procedures and sub procedures
12. Describe how to use dates and timers controls.		12.1	Illustrate how to create and manipulate date variables
		12.2	Define how to execute code at regular intervals using a timer control
		12.3	Analyse how to retrieve date input with a datetimepicker control
		12.4	Illustrate how to use group controls using a groupbox control.
13. Discuss the scope of a variable. Define passing arguments – pass-by-value versus pass-by-reference.		13.1	Demonstrate how to create variables that can be used in all the form's procedures
		13.2	Describe how to pass arguments by reference
		13.3	Define how to eliminate subtle data-type errors by enabling option strict
		13.4	Describe how to change a value from one data type to another, using methods of class convert.
14. Define an array. Understand declaring and allocating arrays. Discuss sorting arrays. Define two-dimensional arrays. Understand RadioButtons.		14.1	Define how to create and initialise arrays;
		14.2	Describe how to store information in an array;
		14.4	Identify how to refer to individual elements of an array
		14.5	Describe how to sort array options in a drop-down list;
		14.5	Determine whether a specific character is in a string
		14.6	Identify how to remove a character from a string
		14.6	Convert a string to lowercase letters
		14.7	Define the difference between one-dimensional and two-dimensional arrays
		14.8	Declare and manipulate two-dimensional arrays
		14.9	Describe the usefulness of two-

		dimensional arrays
	14.10	Demonstrate how to use radiobuttons to enable users to select exactly one option out of several
15. Define sequential access files. Understand files and streams and how to write to a file.	15.1	Demonstrate how to create, read from, write to and update files
	15.2	Describe sequential-access file processing
	15.3	Identify how to use streamreader and streamwriter classes to read from, and write to, sequential-access files
	15.4	Describe how to add and configure a monthcalendar control

Recommended Learning Resources: VB .Net

Text Books	• Beginning VB.NET (Programmer to Programmer) by Thearon Willis, Jonathan Crossland and Richard Blair. ISBN-10: 0764556584 • VB.NET Language Pocket Reference by Steven Roman, Ron Petrusha and Paul Lomax. ISBN-10: 0596004281
Study Manuals	BCE produced study packs
CD ROM	Power-point slides
Software	VB .Net