

CIS159 – Visual Basic I

Lesson Sixteen

Instructor: Gary Marrer

Lesson Sixteen (Final Week)



This Weeks Objectives:

Topics

- Final Exam
- MDI Multiple Document Interface
- ImageList Control
- ToolBar Control
- Outside the Text: Crystal Reports
- Course Conclusion

Week Sixteen

We have reached the end of class. Please refer to Bb for information on the Final Exam and deadlines for turning in assignments. It was a pleasure working with all of you and I wish you much success in your future. If you have any questions on other CIS classes, please send me an email or give me a call.





Lecture

Chapter 14 - Additional Topics in Visual Basic

Multiple and Document Interface

Up to this point in the textbook, we've dealt exclusively with projects that contained references to single forms (**SDI - Single Document Interface**). You may have noticed that one of the things omitted so far from our lecture has been the ability to create applications that have a **Multiple Document Interface - MDI** of multiple forms. Most of the Microsoft products, Visual Studio.NET for example, are set up so that one form acts a container for all other forms. This one container form allows you to close and minimize the entire project with the click of a button. This chapter shows how you could build a MDI application with the one parent form holding one of more child forms. Pay special attention to page 571-574. This section of the chapter describes how to add a window menu option (which also appears on most applications) that will allow you to tile and cascade horizontal the multiple forms contained inside the MDI form.



Toolbar and ImageList Controls

Like MDI, the topic of **ToolBar**, **StatusBar** and **ImageList** controls are also covered as a start off topic and CIS259. Also like MDI, these new controls give us a more professional looking interface. These controls work much the same as the other controls covered of the book but have a couple of twists and turns that make them slightly different in terms of configuration. The **ImageList** control may especially seem out of place since on its own, it really does nothing on the form. We use the ImageList control the supply **collections** of graphics to other controls like the **ToolBar**, **TreeView**, **ListView** controls (to name a few). A **collection** can be thought of as an array of objects. In the case of the ImageList control, it contains a collection of graphic images. The image file formats supported are all the favorites such as .BMP, .GIF or .JPG.

Digital cameras for example typically store pictures in a .JPG (j peg) file format so you now can incorporate pictures taken from your digital camera into applications written Visual Basic.

The **ToolBar** control puts a line of executable icons across the top of your application to provide short cuts to application code that would normally be triggered by a menu option or command button. The toolbar consists of a row of icons (small graphics files). The imagelist control holds in its graphics collection the icons that will be displayed on the toolbar. Unlike ImageList that stays in the tool tray, when you add the ToolBar control, it affixes itself to the form just below the menus. Starting on page 548 of the textbook and continuing through page 551, the textbook demonstrates the steps necessary to add the ImageList and ToolBar controls to the form.



StatusBar Control

The **StatusBar** control is another control that will help make your form look more professional. The StatusBar appears at the bottom of your form and to contain one or many panels that return information about application. You might have a copyright notice, today's date and time or reference information about what is happening with the application (calculating..., loading..., saving..., etc.). The StatusBar is another mechanism for providing feedback back to the user during operation of the program. The StatusBar control can be developed at design-time and/or be modified during run-time. The text book demonstrates adding and configuration the StatusBar at design time. By accessing the built-in MSDN help, you can find additional information about making changes in the StatusBar at run-time.



Outside of the Book but Important: Crystal Reports

Crystal Reports is a standalone product that is long been a favorite of many users for creating reports. It is a standalone product that is normally used as an application product (i.e. MS-Excel) to create links to database for user formatted reports. It is an add-on tool to Visual Basic.NET. Crystal Reports is a very flexible product that can connect to virtually any data source. Although designed more specifically for relational databases (like Access and SQL Express), it can also connect to the text files what we generated in Chapter 11. As you look at

Crystal Reports, you see that covers a number of different report formats along with even a charting capability.

Crystal Reports works a lot like the report generator in Microsoft Access. It allows you to customize report templates and specifically customize the different report bands contained within that report. Each band represents part of the report. There's a band for the header, there's a band for the details section of the report and there is a band for the footer section. There are additional bands provided depending on level of grouping and control break processing that the report needs to support. Fortunately, there is interface/wizard that walks you through the steps of connecting to data source, designing a report, and assigning a sort and grouping specification. In the CIS259 class, you will learn how to connect ADO DataSets to your reports. Once a report template is defined, you will drag from the toolbox the Crystal Report Viewer to that view the report template.



Book Notes:

I would like any feedback you have on the book via an e-mail. I am especially interested to know if the book was appropriate for they hybrid learning format.



MY Turn:

Course Conclusion

This class has all of the same material, labs and tests that a traditional classroom version of CIS159 has. Whether this made it harder or easier is probably dependent on the student. It was certainly equal. Stick with it and practice Visual Basic. Take a Java course and improve you understanding of object orientated programming or maybe take the CIS259 (Adv. Visual Basic) and learn more advanced topics.

I am very interested in any feedback you might have on BlackBoard as a learning tool, the coverage of Visual Basic topics, the presentation and content of the lecture.

Good luck to all of you.



Chapter Review

(1) What is the difference between SDI and MDI?

(2) Why is the imagelist control required for the toolbar control?

Questions?