**Chapter 5 : Strings and Table Construction (pgs. 150-209)**

*In this chapter you will learn to manipulate strings, menus, loops and scrollbars.*

Section One: Strings

What is a String?

A string is a **series of characters***.* Examples are: digits, letters, punctuation and special characters (@, #, $, etc.). Properties of a string are its length, its ability to be concatenated, searched, displayed and read.

**concatenation:** (in programming) joining two strings together

In Visual Basic you can concatenate strings using the & (ampersand). Ex.

strFirstName = "Sally"

strLastName = "Parker"

strFullName = strFirstName & strLastName

In the above case "Sally" is a **string literal**and strFirstName is a **string variable**.

Strings can be compared using the =, <, >, and <> signs. Visual basic compares strings based on their **ASCII value*.***

**ASCII value:** The American Standard Code for Information Interchange (acronym: ASCII) codes represent text. Lowercase letters range from 97 to 122 in the ASCII code and uppercase letters range from 65 to 90.

String Functions (see pg. 156 -- 160 for more information)

**UCase** and **LCase** - convert all characters of a string to uppercase or lower case respectively

**StrComp** - compares two strings

**Len** - counts the number of characters within a string

Section Two: Menus & MiniEdit

So far you have used command buttons for the user to choose a command. Another way to do this is by using the menu bar, you will find that this can streamline the look of your application.

Menu buttons also utilize the Click event. You can create a menu using the menu design window in VB 6.

* · Be sure to transfer the code to the menu buttons
* · if the program originally ran using command buttons---delete the buttons
* · remove methods such as 'SetFocus' that are no longer relevant.

**Project MiniEdit**: This project will help you to construct a simple word processer. Here are a few functions you will need to build this project.

**Inputbox** - collects information user enters from keyboard, returns string, syntax:

**Trim$(str)** - takes a string as a parameter and removes all leading and trailing spaces

**UCase$(str)** - takes a string and converts to uppercase

**LOF** - takes parameter filenumber, returns number of bytes in file

**Input$(filesize, filenumber)** - reads a string of length filesize from file with given file number (see handout on file input output)

Section Three: For-Next Loop

The **For-Next** loop is a **definite loop*.*** In a definite loop its is known how many times you want to execute the code within a loop.

A loop allows you to repeat a section of code. It is a common control structure.

Sample syntax:

For i = 1 To 100 'Step 2 (increment)

val = val + 1

list1.additem (val)

Next i

In this case, For, To, Step and Next are reserved words. The above code increments val by 1, 100 times and outputs it to listbox. The step section of the loop is commented out, step allows you to control at which value of the looping variable (i) you execute the code. Step 2 would mean that every time the value of i increments by 2 the code within the loop will execute.

You can nest For-Next loops within each other as you learned to nest If-Then statements. Essentially a loop within another loop.

Section Three: Scrollbars

Scrollbars come in two versions: horizontal and vertical. Each includes a movable scroll box and two scroll arrows at each end. The scrollbar can be used to display output, show progress or position the cursor in a textbox.

The parts and pieces of scrollbox are best exampled using a diagram (see pg. 183). Here are some object properties you need to know.

**Value** -- the current position of the scroll box

**Min** -- sets starting value of the Value property

**Max** -- sets the maximum value of the Value property

**SmallChange** - controls rate of motion

Additional information from this Chapter:

* Constants : are values that cannot be changed while a program runs, unlike variables, **Const** is the reserved word to declare a constant, ex. Const PI = 3.141592654
* Each item of a listbox has a number. The first is item 0. The ListCount property shows the number of items in the list