Tutorial Sheet 3 /Week 4

IMS1906: Business Software Fundamentals

Tutorial exercises

Week 4: Coding a random number generator with simple mathematical calculations

These notes are available on the IMS1906 Web site  http://www.sims.monash.edu.au

Please contact your tutor if you require assistance with these exercises. Tutor email address and consultation times are available on the subject’s Web page under Staff.

Aim

To introduce students to:

• classes Convert and Random
• assigning properties to a control at design-time and run-time,
• in-built VB Method: Focus()
• in-built VB functions: Val() and Format()
• formatting controls (ie. Align buttons, make same size) and access keys.
• TOE charts

Task

To create a Random Number Generator with supporting mathematical operations (Addition, subtraction, division and multiplication)

Requirements

1. Develop a TOE chart to plan the application.
2. Design the calculator in the Windows Form Designer Window and write the code.
3. Make an executable version of the application, test it and run it for your tutor.

Description

Design an application that allows a user to generate two random numbers between 1 and 100 and display them in two different text boxes. Once the random numbers have been generated allow the user to choose (by clicking the appropriate command button) a valid mathematical operation (+, -, /, *) and perform the operation on the two random numbers generated. The Clear command button removes the numbers in the text boxes, and the contents of the labels containing the operator and the answer. Exit ends the application.
Part 1: Develop a TOE chart to plan the application (do this before lab)

1. Draw a TOE chart with three column headings Task, Object and Event
2. Identify the application’s tasks
3. Identify the objects which you will assign to those tasks
4. Identify the events required to trigger an object into performing its assigned task
5. Design the user interface

Part 2: Coding the behaviour

To generate a random number you need to declare a random object from Random Class, as demonstrated below:

```vbnet
Dim randomNum As New Random    'declare random object
txtNum1.Text = Convert.ToString(randomNum.Next(1,101))    'generate random number
txtNum2.Text = Convert.ToString(randomNum.Next(1,101))    'generate random number
```

The example above produces a whole number from 1 to 101, including 1, but excluding 101.

- Use the following controls: two text boxes (name them `txtNum1` and `txtNum2`), three labels (name them `lblOperator`, `lblEquals` and `lblAnswer`), and 7 command buttons (name them `btnRandom`, `btnPlus`, `btnMinus`, `btnTimes`, `btnDivide`, `btnClear` and `btnExit`).
- Use Access Keys by putting an & in front of the letter in the text property of the Exit and Clear Buttons
- Align text boxes and buttons, ensure button of similar functionality are of the same size.

**VB in-built Method Focus**

The **Focus Method** allows you to move the focus to a specified control while the application is running. You can also improve your application by using the focus method to return the cursor to the first text box after the calculator has been cleared.

**VB in-built Functions: Val() and Format()**

Remember to use the in-built `Val()` function to convert the text within each text box (a string) into a numeric value (since mathematical operators work on numbers only). Use the `Format()` function to make the appearance of your application better, by displaying the result with two decimal places.

Part 3: Test, Run and Debug your application

Run and test your application.

**Exercise 2**

*Multi-choice Questions from Zak p65 [Q1, Q2, Q3, Q4, Q10, Q11, Q13, Q15, Q18]*

*Coding question Zak p69 [Crispies Bagels and Bites]*