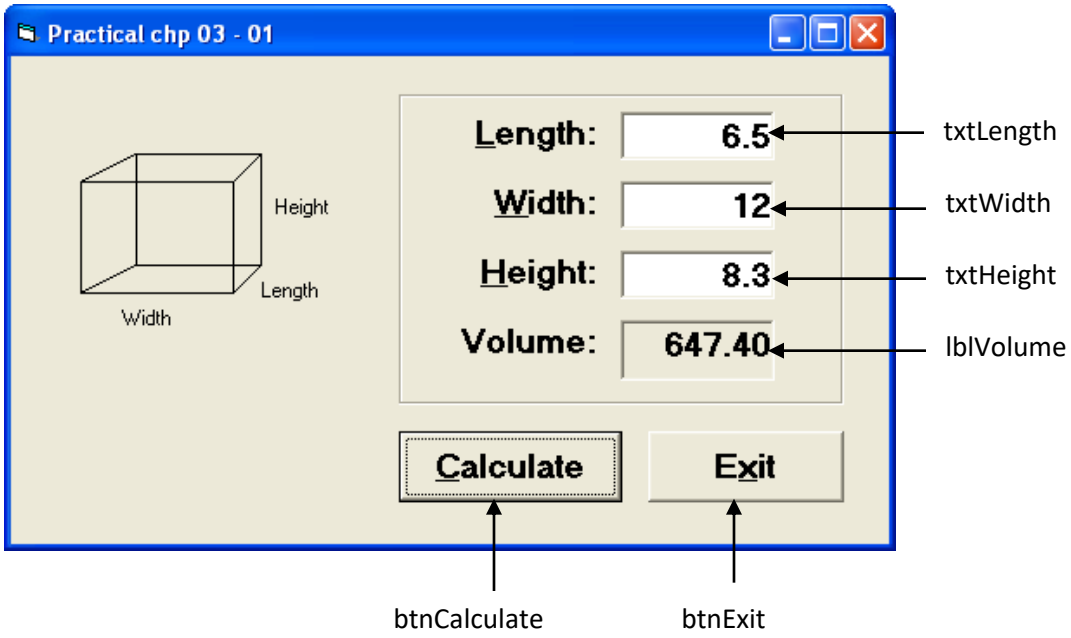
Practical 3: Variables, Constants, and Calculations

NOTE: Turn on Option Explicit and Option Strict at the project level.

Q1.



Create the above project that based on the following requirements:

a) Design and set the objects properties for the user interface as shown above.

b) Allow the user to enter the length, width and height for a cube object in particular text boxes.

c) Set the border style property for *lblVolume* to fixed single during design time.

d) Improve the usability of the program by setting appropriate tab order, shortcut keys, default focus, accept and cancel buttons, etc.

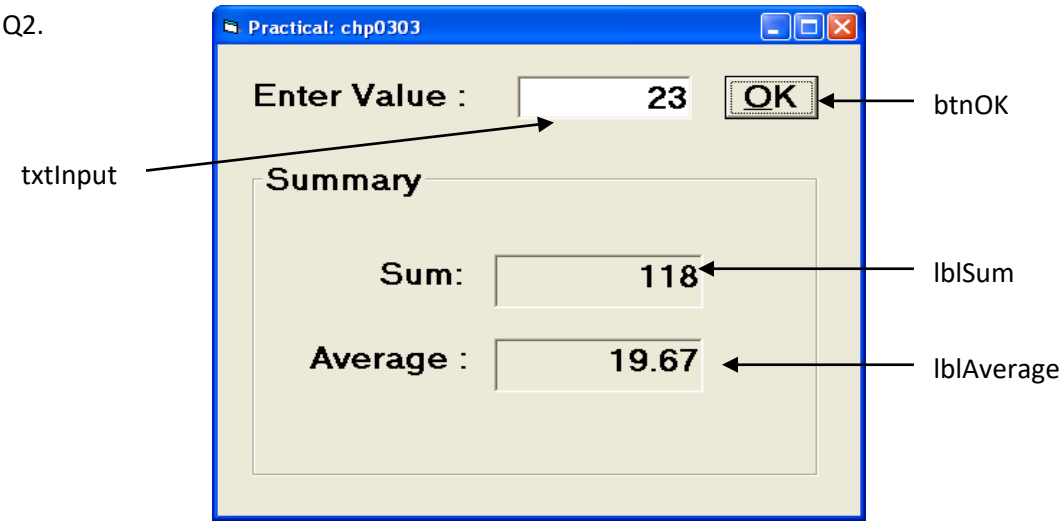
e) Calculate the volume for a cube object when *btnCalculate* button has been clicked. Format the result into two decimal points.

[Hint: volume = length * width * height]
NOTE: Use **try-catch** block to handle input/conversion exception.

f) Terminate the program when *btnExit* button has been clicked.

g) Test and save your project.

Q2.



Create the above project that based on the following requirements:

a) Design and set the objects properties for the user interface as shown above.

b) Allow the user to enter value in *txtInput* box.

c) Set the border style property for *lblSum* and *lblAverage* to fixed single during design time.

d) Improve the usability of the program by setting appropriate tab order, shortcut keys, default focus, accept and cancel buttons, etc.

e) Calculate the sum and average value when *btnOK* button has been clicked.

[Hint: sum = sum + input value, average = sum / number of inputs]

NOTE: Use class-level variables to remember the accumulated sum and input count. Use **try-catch** block to handle input/conversion exception.

f) Test and save your project.

Q3. Create a project that will input an employee's sales and calculate the gross pay, deductions, and net pay. Each employee will receive a base pay of RM900 plus a sales commission of 6 percent of sales.

After calculating the net pay, calculate the budget amount for each category based on the percentages given.

Pay

Base pay	RM900; use a named constant
Commission	6% of sales
Gross pay	Sum of base pay and commission
Deductions	18% of gross pay
Net pay	Gross pay minus deductions

Budget

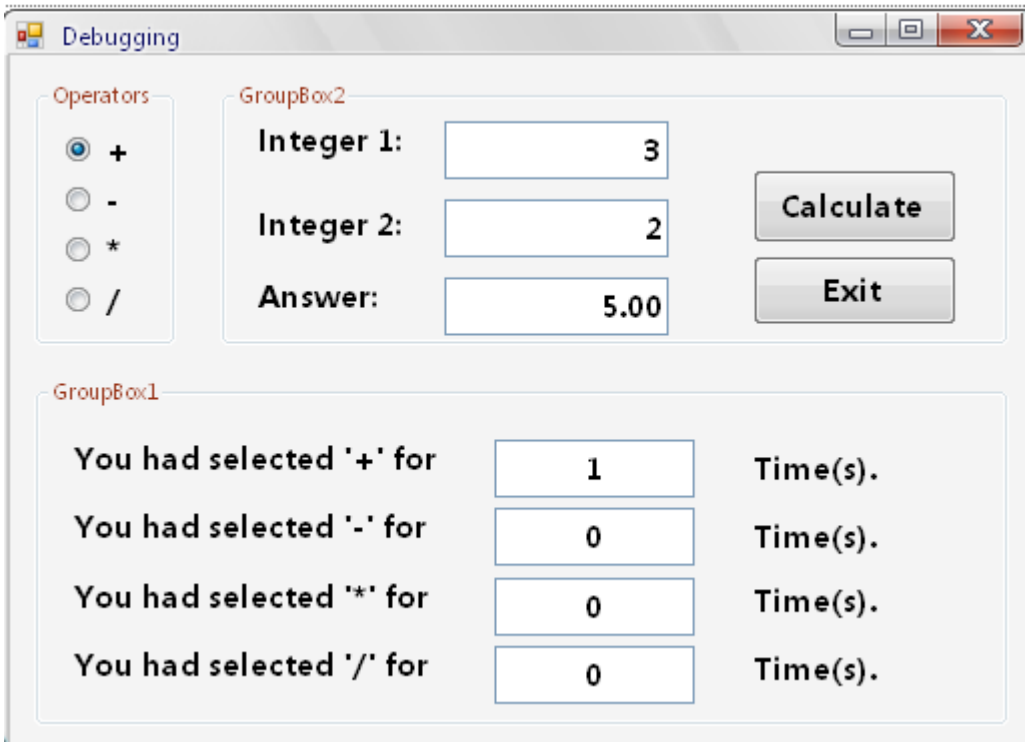
Housing	30% of net pay
Food and clothing	15% of net pay
Entertainment	50% of net pay
Miscellaneous	5% of net pay

Form: Create a form with appropriate controls and usability features to handle the scenario above. Use text boxes to input the employee's *name* and the *sales amount* in RM. Use labels to display the result of the calculations.

NOTE: Use **try-catch** block to handle input/conversion exception.

Provide buttons for *Calculate*, *Clear*, and *Exit*. Display a message to the user for any invalid input data.

Q4. Debugging Skill



Open the given project named **P3Q4_Stud**. The form **FrmDebug** (as shown above) cannot be compiled and executed as it contains some errors and bugs. Debug the form.

The form performs the following tasks:

(i) *btnCalculate*: Calculate and display the answer on lblResult based on the operator selected by user, and display number of times the particular operator had been selected (i.e. to accumulate the counts).

(ii) *btnExit*: Terminate the program execution.

NOTE: DO NOT turn off Option Explicit and Option Strict.

Once you solved all the errors and bugs, modify the form by improving its usability.