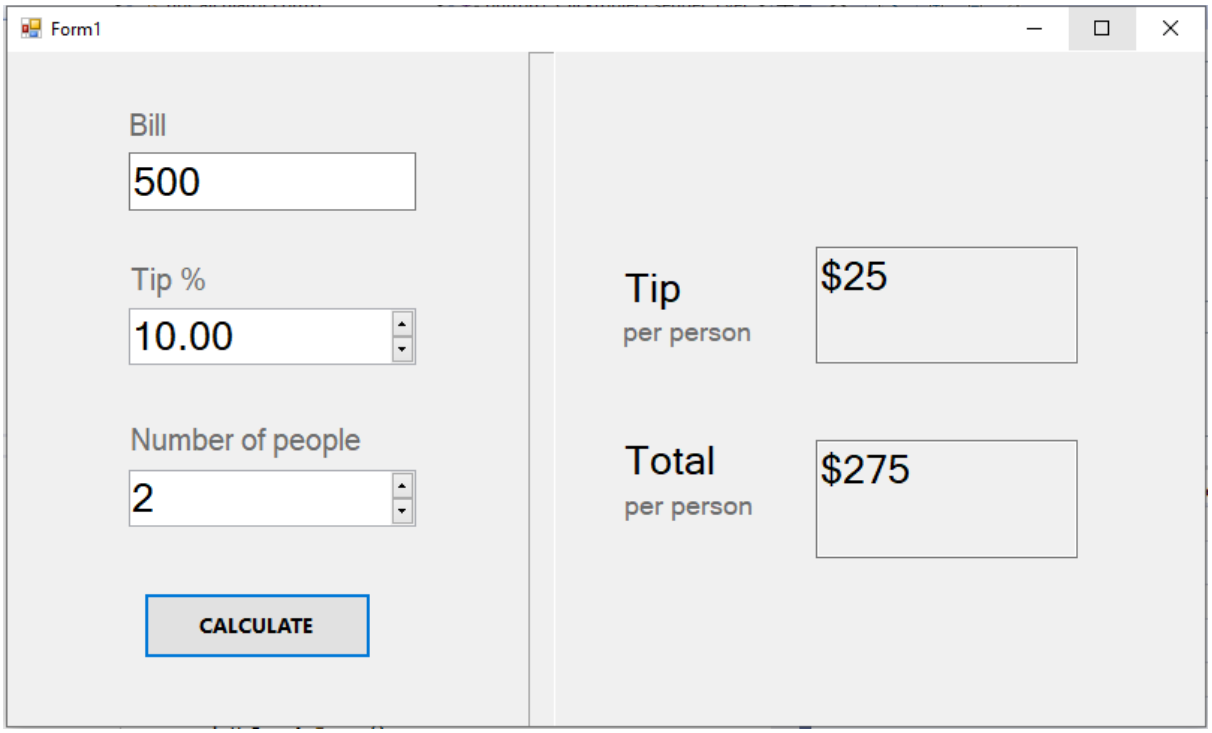
GrapeCity Assignment

Project Name: Tip Calculator

By: Vanshika Bansal, Shiv Nadar University

TEST CASES:

- ✓ UI Interactions: The user can interact with:
  - I. {Bill} amount textbox.
  - II. {Tip%} textbox along with up-down buttons.
  - III. {Number of People} along with up-down buttons.
  - IV. {Calculate} button.
  - V. {Minimise, Maximise, Cross} buttons on top-right.



- ✓ UI Testing:

TC1: Check if the labels and corresponding text boxes should be mapped correctly. The user should not get confused regarding the association of labels with boxes.

TC2: Font - Check if the text is readable with user-friendly font style, not over-crowded or isolated. The font colour and size should represent the hierarchy and importance of labels.

TC3: Language - Since the application is used in different parts of the world, check if the presented language and currency corresponds to the standards of your respective country.

TC4: Check if the maximise, minimise buttons work and does the cross button closes the application without running into any problems.

TC5: Bill, Tip%, Number of People - Check if an error message is displayed if {calculate} button is clicked while the textbox is empty. Similarly, an error should be displayed if the input is not numerical (e.g., a, *, + etc).

TC6: Tip%, Number of People - Check if the textbox contains default values. Tip% should be 0.00 while Number of People should be 1 by default.

TC7: Tip%, Number of People - Check if the up-down buttons are working, also the user should be able to type the digits as an alternative against up-down functionality.

TC8: Calculate - Check if {calculate} button is working fine. It should display the results if everything is fine else, a suitable error message is displayed if some fields are unsatisfying.

TC9: Tip Per Person, Total Per Person - If the program runs correctly, the output fields should display numerical results (with respective currency) having a precision up to 2 decimals.

TC10: Tip Per Person, Total Per Person – The output should be read only so that no one can tamper with the output data.

- ✓ Functionalities: The user enters values in the respective 3 textboxes to generate the tip and bill amount results. Here, we check the valid data by creating classes on input values. These classes can be categorized as (+ve value), (+ve value), (1 to 100), (0) where (+ve value) includes (0) and (1 to 100).
  Note that the admissible values of Tip% are 0 to 100. (Max percentage is 100%)
  Note that the admissible values of Number of People are 1 to 100. (A limit of 100 people)

- ✓ Functionality Testing:

Case No.	Bill input	Tip% input	Number of People input	Results
TC11	-ve value	Any value	Any value	Error message displayed
TC12	0	Any value	Any value	Result is always 0
TC13	+ve value	-ve value	Any value	Tip% changes to 0.00 which is the minimum admissible value
TC14	+ve value	0 to 100	Any value	Results work fine
TC15	+ve value	More than 100	Any value	Tip% changes to 100.00 which is the maximum admissible value
TC16	+ve value	0 to 100	-ve value	Number of people changes to 1 which is the minimum admissible value
TC17	+ve value	0 to 100	0	Number of people changes to 1 which is the minimum admissible value (since division by 0 is logical error)
TC18	+ve value	0 to 100	1 to 100	Results work fine
TC19	+ve value	0 to 100	More than 100	Number of people changes to 100 which is the maximum admissible value
TC20	+ve value	0 to 100	1 to 100	Results work fine