Homework 2 - Multiplication Tables

Due: Monday, March 7th @ 11:59pm

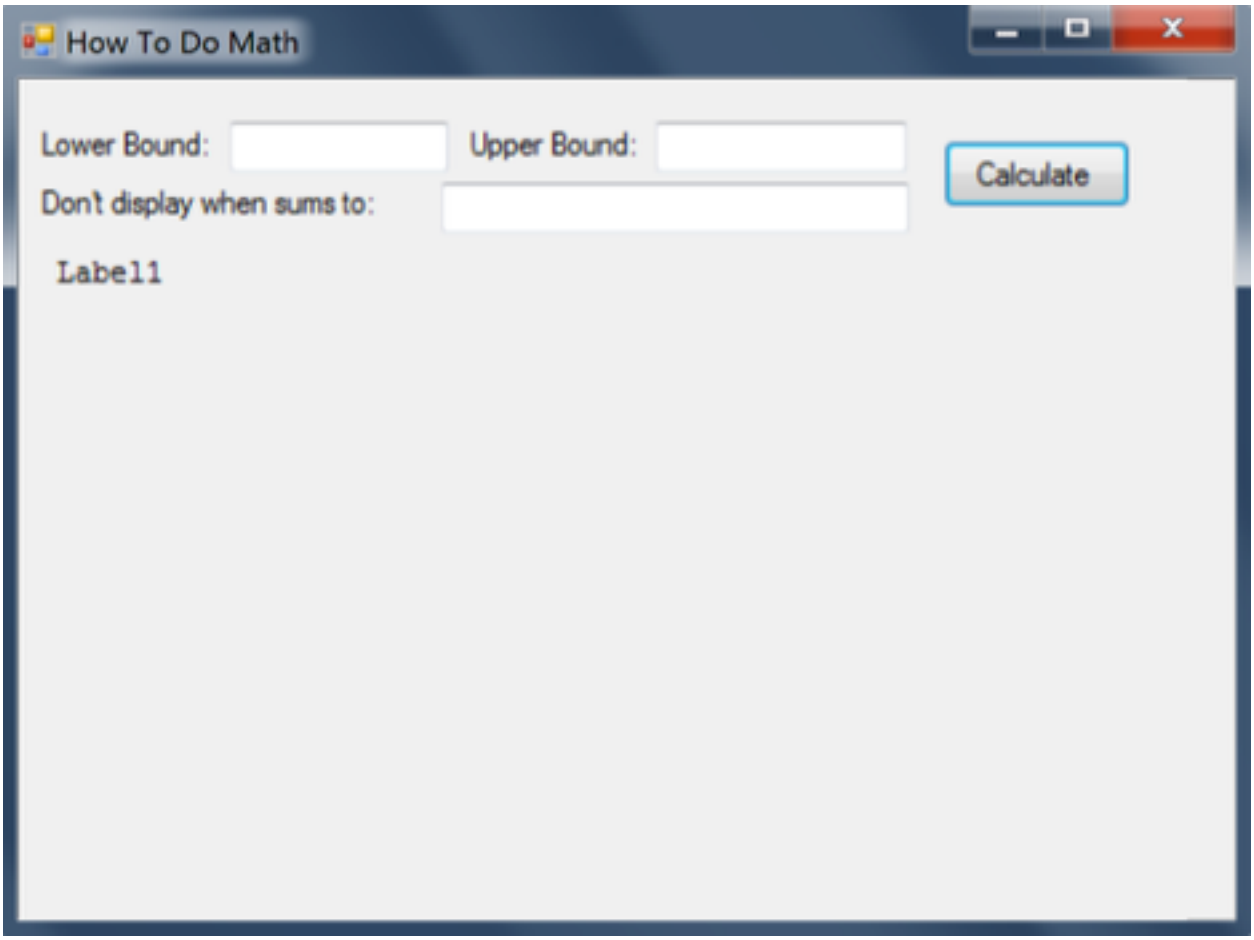
Your sister is such a pain. She is just learning basic math but can't quite grasp the idea of multiplication. Every few minutes she will come up to you and ask you what the product of two numbers is. Finally you decide to give her ALL the numbers.

Using your amazing Visual Basic skills, you will make a program that displays the multiplication table for a range of numbers. To do this you will need to use at least one loop.

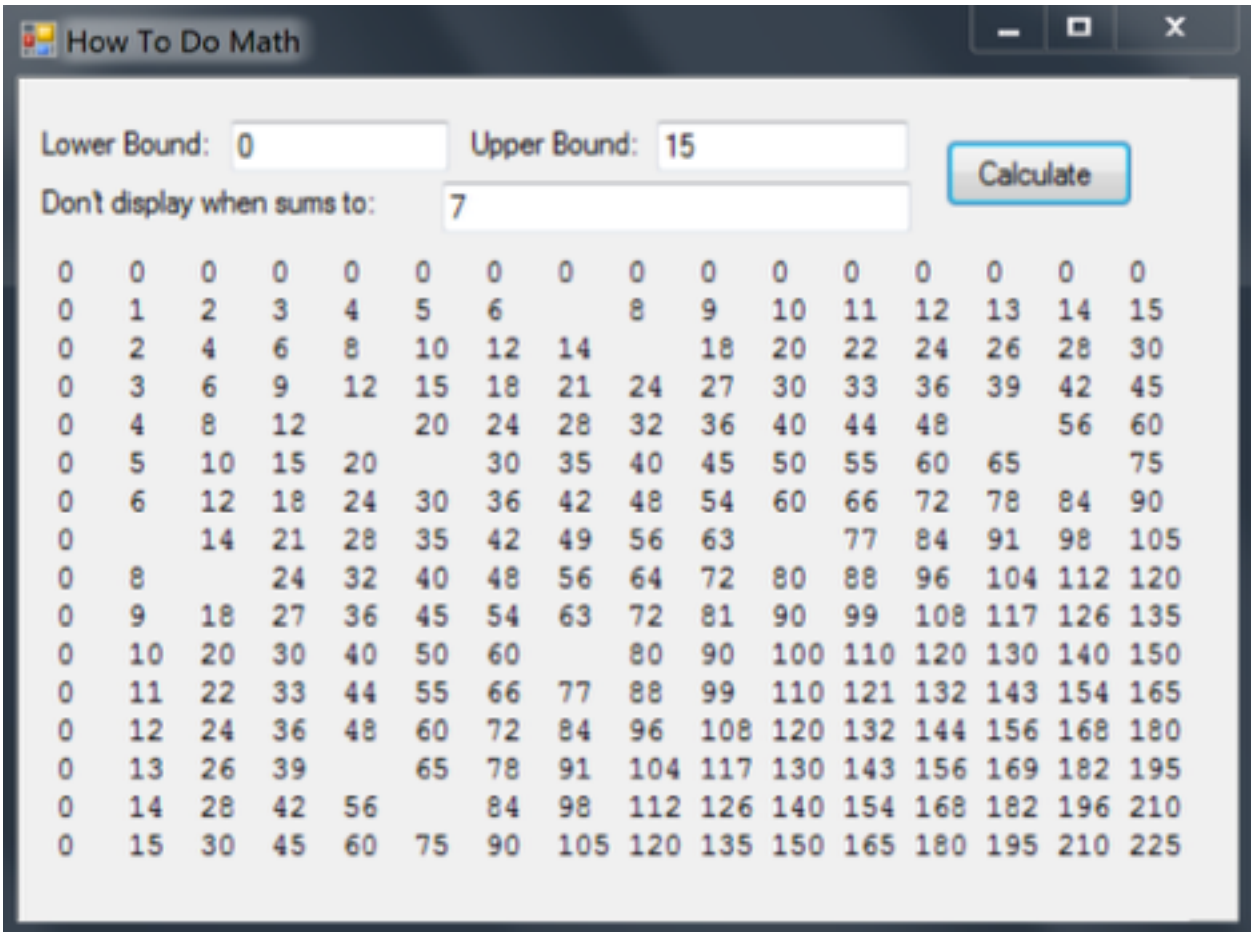
For example if the user chooses 2 as the lower bound and 5 as the upper bound the table should look like:

```
4   6   8   10
6   9   12  15
8   12  16  20
10  15  20  25
```

Recreate the GUI below.



Which when you are done will create something like this:



Things of note:

- Make sure Option Strict is On and Option Explicit is On.
- There are only 4 labels total. Three as labels for the text boxes and one to display the multiplication table.
- In order to use only one label like this you will need to “concatenate” strings together. This is like adding but for strings. `1 + 2 = 3` but `“1” & “2” = “12”`
- At then end of each row concatenate on a vbNewLine to start the next row. Try this command: `Label1.Text = “First Line” & vbNewLine & “Second Line”`
- You ***must*** set the font on the multiplication table label to a monospaced font such as `Courier New`. A monospaced font is one where every character is the same width so that they line up in columns nicely. Monospaced fonts are ubiquitous in programming, the Visual Studio code editor uses a monospace font.

- You should try to get the columns to line up nicely. The best way to do this is to use padding. For example, the string “Q” is exactly one character long. myQString.PadRight(4) is at MINIMUM four characters long. If spaces were pound signs, this function would return “Q###” making the total length 4. I would suggest you use PadRight(4) so that you have 3 spots for digits and one extra space as the actual space between the columns.
    - For extra credit, pad the correct amount based on the length of maximum product. For example, for 15*15=225 so the correct padding is 4 (3 digits + 1 for spacing). For 100*100=10,000 so the correct padding is 6 (5 digits + 1 for spacing).

Why are there blank spots?

Because your sister is picky (or something, just do it).

You will **NOT** print numbers whose components sum to some other number you input. For example if your “don’t print when sums to this” number was **9** then you wouldn't print numbers such as 9, 90, 18, 81, 225, 180, etc.

Things to explore to figure this out:

Modulo operator. This allows you to find the remainder. (VB uses mod)

Ex. 25 mod 10 = 5

Integer division. This allows you to ignore decimals. (VB uses \ )

Ex. 25 \ 10 = 2

You only have to handle “not printing” up to three digit numbers. For extra credit try to handle numbers of arbitrary length. If you DO decide to do the extra credit, please comment your code explaining how you did it.