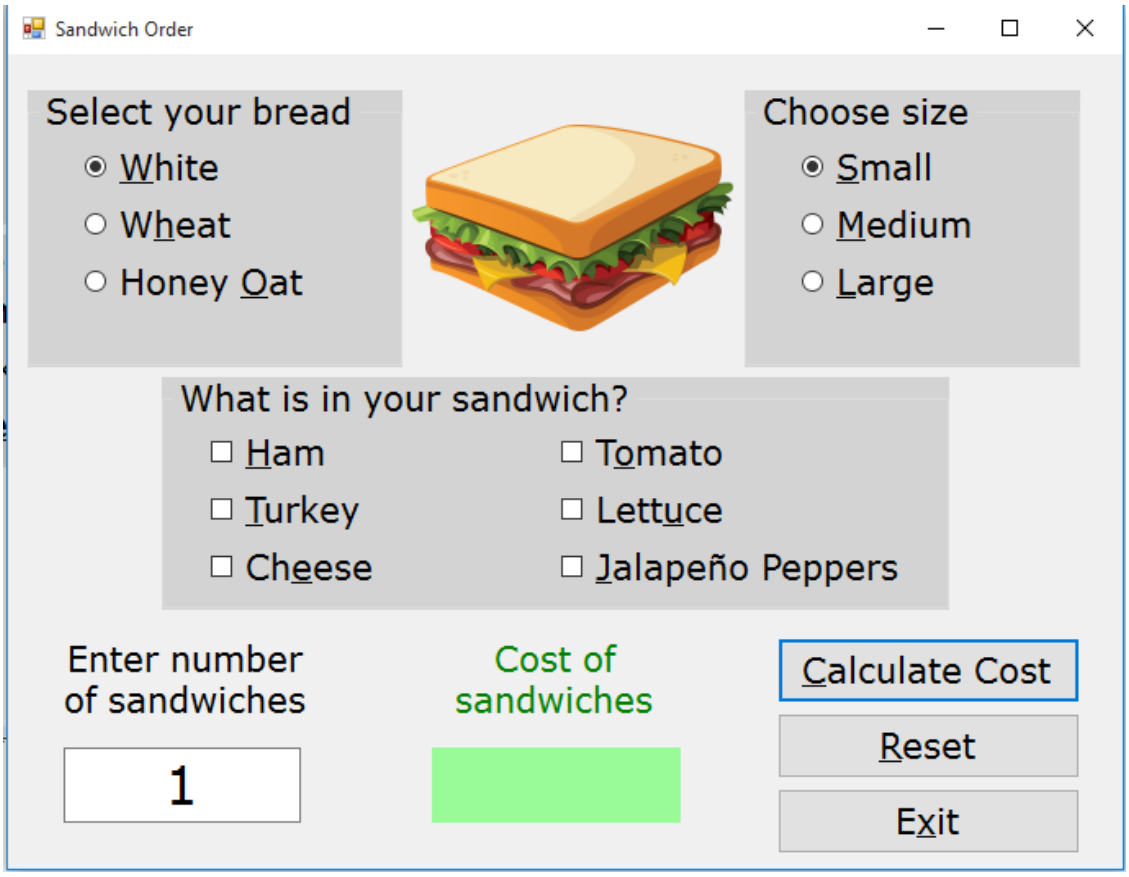
Lab 4B – Sandwich Order

You will create a program involving GroupBoxes, RadioButtons, and CheckBoxes.



Users will select exactly one type of bread, exactly one size for the sandwich, and as many items as they would like inside the sandwich. They must also choose how many sandwiches of this type they would like (the default is one sandwich). When a button is pressed to calculate the cost (don't need to worry about sales tax, tip, etc.), the price of the sandwiches is displayed.

You may use something besides a sandwich (or other sandwich options) - pizza (crust type, diameter, and toppings) and ice cream sundaes (containers- bowls, cones, etc., size, and toppings), for example. Be creative - it doesn't have to be food-related. The program must include

- one category from which there are at least three items to choose, and only one may be selected
- a second category from which there are at least three items to choose, and only one may be selected
- a third category from which there are at least six items to choose, and users may select as many as they wish
- each item has a cost/price associated with it

Objectives

- Validate input through parsing
- Use of GroupBoxes
- Use of RadioButtons
- Use of CheckBoxes

Requirements

- Comment your code!!!

- Whatever you use (sandwich, pizza, ice cream sundaes, etc.), follow the details given on page 1 about how many items can be selected from each category. This will determine whether you use RadioButtons or CheckBoxes for a category.

- All of the prices are fixed values and should be declared as constants. You may use your own prices - here are the values used for the samples at the end of this document:
  - White bread - $0.10
  - Wheat bread - $0.10
  - Honey Oat bread - $0.20
  - Small sandwich - $2.49
  - Medium sandwich - $3.49
  - Large sandwich - $4.49
  - Ham - $1.25
  - Turkey - $1.45
  - Cheese - $0.50
  - Tomato - $0.20
  - Lettuce - $0.10
  - Jalapeño Peppers - $0.25

- Format the output as currency

- Instead of using a Try-Catch block to validate what the user puts in the textbox, use `Integer.TryParse(the text property of the textbox, the integer variable into which you want to put the converted text if it works)`. This is explained in your book on pages 252-255, including Tutorial 4-8 (all the code is written for you in this tutorial). Display a message if users don't have appropriate data in the textbox.
- You can also view a video about the TryParse method (based on a program available at http://bowerpower.net/compprog1/vb04/TryParseMethods.zip) in one of these three ways:
  - Screencast: http://v2.screencast-o-matic.com/u/etC7/TryParseMethods
  - YouTube: https://youtu.be/z0ZsKDYrr1c
  - Download the MP4 file: http://bowerpower.net/compprog1/vb04/TryParseMethods.mp4

- When you are checking which one of a group of RadioButtons has been checked (only one can be checked), you will use `If…Else If`.

- When you are checking which of a group of CheckBoxes has been checked (more than one can be checked), you will need to use a series of individual `If` statements.

- You can assign tab order to RadioButtons in a GroupBox and CheckBoxes in a GroupBox. This is explained (sort of) on pages 258-259 in your book. Mr. Bower may make a video explaining this – if he does, this document will be edited to feature a link; if he doesn't, ask him to show you how it is done when you're ready to check it out!

- When users change the data in the textbox, the result in the answer label should go away (use a TextChanged event to address this).

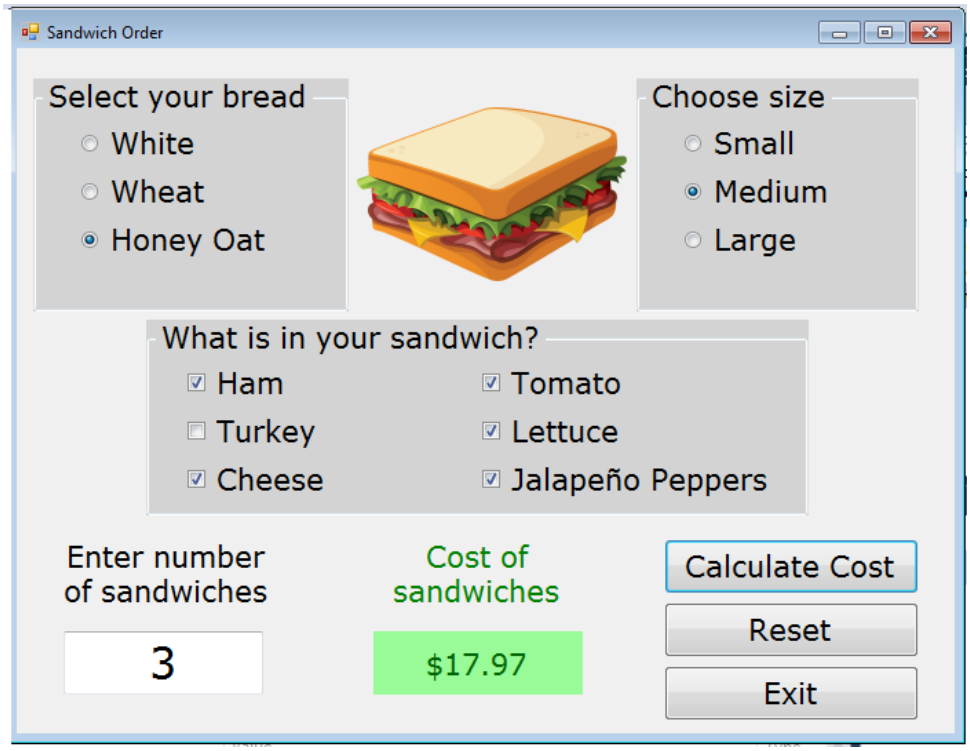
- Radio Buttons and CheckBoxes should have access keys (as well as the buttons). Remember to use the ampersand (&) in the text property just before the character you wish to serve as the access key. You don't want to repeat any access key letters and you will have at least fifteen of these on your form!
- You can get some extra points if you choose to incorporate invisible RadioButtons. We can make those the checked RadioButtons at the start (and when the program is reset) and users think no RadioButtons are checked.
- You can view a video about invisible RadioButtons (based on a program available at http://bowerpower.net/compprog1/vb04/InvisibleRadioButton.zip) in one of these three ways:
    - Screencast: http://v2.screencast-o-matic.com/u/etC7/InvisibleRadioButton_RadioButtonClickEvent
    - YouTube: https://youtu.be/LrVkkDQ9BUs
    - Download the MP4 file: http://bowerpower.net/compprog1/vb04/InvisibleRadioButton_RadioButtonClickEvent.mp4
- If you use invisible RadioButtons, you will need to make sure users actually selected visible RadioButtons before the cost is calculated – if they didn't, you'll need to give them an error message and not make any calculations.

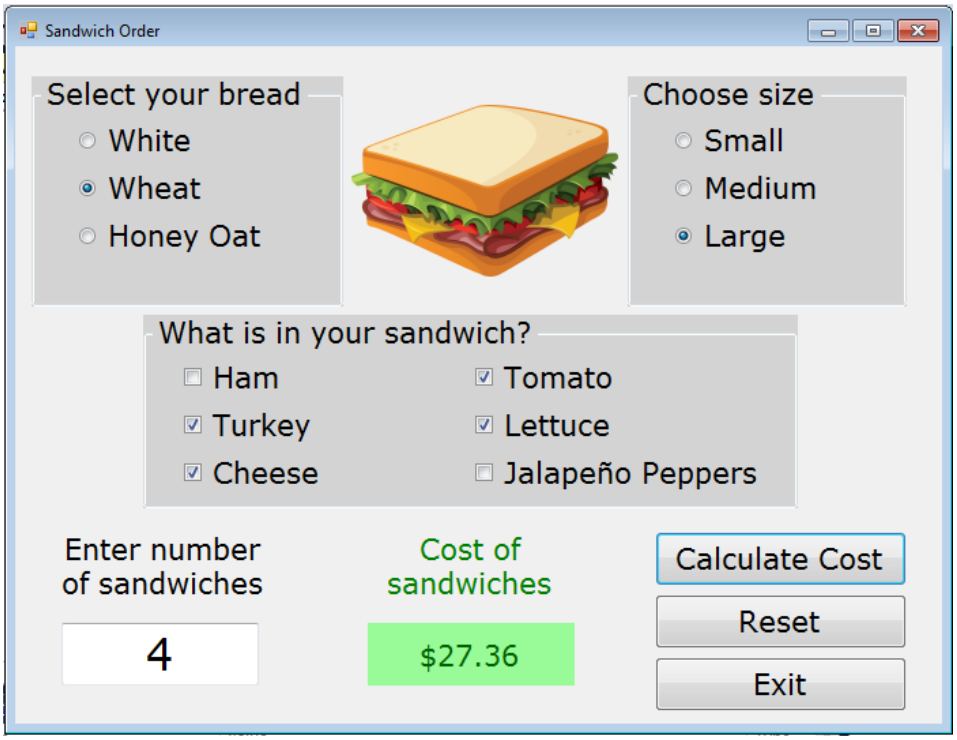
- Type and modify the following at the top of the code:

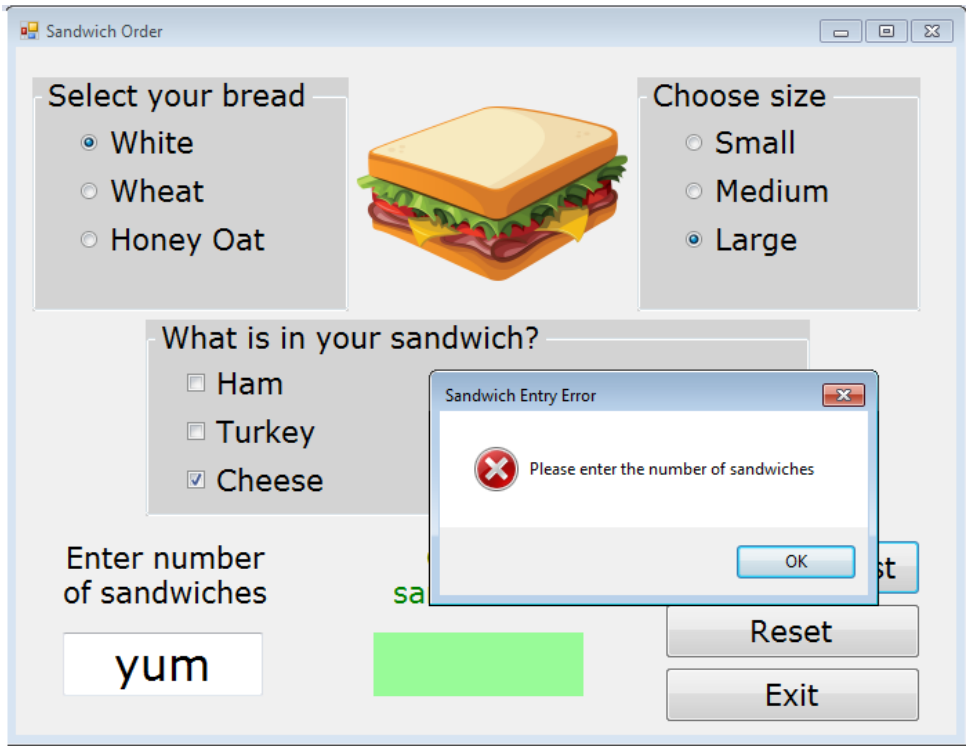
```
'Your Name
'Date
'Assignment Name (Lab 4B – Sandwich Order, for example)
'Computer Programming I - Bower (or CS 11400 – Bower if you want IPFW class)
'This program will... (describe the program)
```

*** For some reason, when you assign two RadioButtons as checked as you design the program (one RadioButton in each GroupBox), only one RadioButton is shown as checked, even though you expect to see two (one for each GroupBox). You'll have to run the program to see if you've got what you want at the start.

Sample Data


Sandwich Order
Select your bread
White
Wheat
Honey Oat
Choose size
Small
Medium
Large
What is in your sandwich?
Ham
Turkey
Cheese
Tomato
Lettuce
Jalapeño Peppers
Enter number of sandwiches
3
Cost of sandwiches
$17.97
Calculate Cost
Reset
Exit


Sandwich Order
Select your bread
White
Wheat
Honey Oat
Choose size
Small
Medium
Large
What is in your sandwich?
Ham
Turkey
Cheese
Tomato
Lettuce
Jalapeño Peppers
Enter number of sandwiches
4
Cost of sandwiches
$27.36
Calculate Cost
Reset
Exit



If you use invisible RadioButtons (images to come soon!)…

Grading (will be completed later)

TOTAL	**20**