#### Due: By the end of the lab session

# Objectives

* Selection
  + Cascading If
* Named Constants
* Scope Rules - Counting

# Program

Design a C# Windows Form Application to take orders in a local pizzeria. The user enters the number of pizzas and selects the toppings from the ComboBox. Assume that all the pizzas in one order are of the same type. The program then computes the price of this order. Assume that the price of a plain pizza with no toppings is 6.00. There is an additional charge per pizza based on the topping provided in the ComboBox, see the table below.

GUI Design – Design a GUI similar to Figure-1. Follow naming standard to name the objects placed on the Form. You may use a *PictureBox* control to add the image of a pizza as well. It is not required.

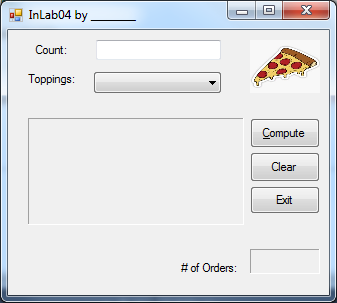
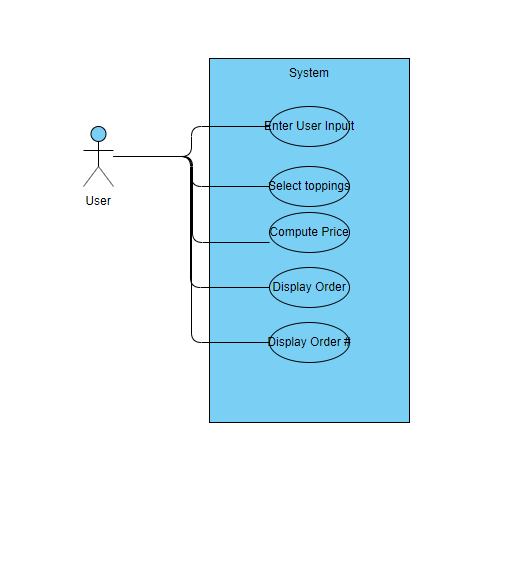


Figure Screen capture when Compute button is clicked.

* **ComboBox:** cboTopping
  + Items**:** Meat Lovers, Pepperoni, Mushroom, Sausage, Veggie, Plain.
  + Set the Dropdown style to **DropDownList.**
* **Class Scope Declarations:**
  + Declare a counter to count the number of orders (NOT pizzas)
  + Declare a named constant for the price of a plain pizza = $6.00. Note: Price of each pizza is price of a plain pizza + price of the selected topping.
  + Declare the following constants (in the table below) with module scope as named constants- Prefix the name of the constants with lower case **c.**

|  |  |
| --- | --- |
| **Type of pizza** | **Extra cost per pizza** |
| Meat Lovers | $ 4.5 |
| Veggie | $3.5 |
| Pepperoni | $ 2.0 |
| Sausage | $ 2.50 |
| Mushroom | $1.00 |

* **Diagrams:**
  + Use Case Diagram
  + Activity diagram to show the logic behind the Compute button.
  + Snip, paste the diagrams to a Word document, and submit the document to the inlab link.



* **Compute Button:**
  + Declare a variable (String) to store the selected topping.
    - Store the selection from the ComboBox in the variable.
  + Declare an integer variable to store the number of pizzas ordered.
  + Declare variable(s) to store price of one pizza, and total price of the order.
  + Use a *cascading if statement* to compute the price of one pizza based on the selected topping – use the named constants in calculation.
    - NOTE: Price of one pizza is = price of the plain pizza plus the price of the selected topping.
  + Compute the total price by multiplying the number of pizzas by the price of one pizza. Display the due price in Currency format as shown in Figure-2.
  + Update the number of orders – display it in the small Textbox.
  + If the number of orders is equal to 10, display Congratulations!! at the end of the output box as shown in Figure-3.
* **Clear Button:**
  + Clear the Textboxes, ComboBox and the output box.
  + Set focus to the top text box.
* **Exit Button:**
  + End the program execution.

# Introductory Comments

‘=========================================================

‘ Your Name

‘ Email:

‘ CNIT 155 InLab-04

‘ Program Description: A brief description of the program in your own words.

‘===========================================================

|  |  |
| --- | --- |
| Figure-2 Sample screen capture | Figure-3 Screen capture when 10th order is processed. |

# Submission

Before submission, test your program by entering different data values and click on all buttons and inspect the output.

* To get full credit zip the outer folder of the project and submit the zipped folder electronically to Blackboard by clicking on the InLab link.
* Submit the document with your diagrams to the same link.