**Terms to review:**

literal string

literal constant

identifier

variable

constant

declaration

initialization

assignment

argument

data type

block

scope

control statement

conditional operator (aka. ternary operator)

switch structure

nesting

truth table

not operator

short-circuit evaluation

range check/match

***definite loop***

***indefinite loop***

***infinite loop***

***iteration***

***accumulator***

**Homework & Labs**

*// Please name your projects LB1, LB2, LB3, etc*

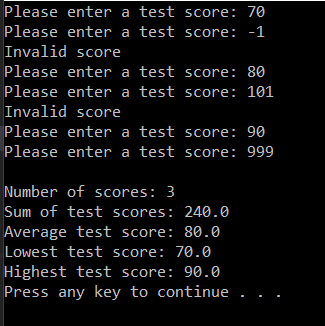
LB1. Complete Naming Conventions Handout

LB2. Complete Data Types Handout

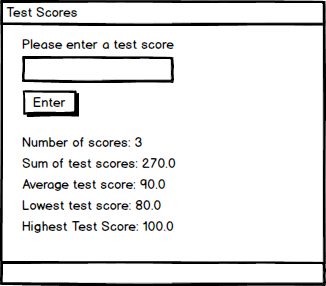
LB3. Complete Prefix/Postfix Handout

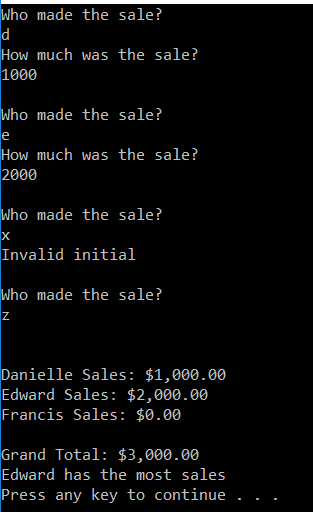
LB4. Write a console application that accepts an indefinite number of test scores for a student, and then computes some statistics about those scores:

* Prompt the user for test scores until they enter 999
* If the user enters a score inside the range of 0 through 100
  + Accept the score
  + Prompt the user for another score
* If the user enters 999
  + Ignore the test score
  + Do **not** display an error message
  + display the number of scores, sum, average, min, and max
  + stop the program
* If they enter a score outside the range of 0 through 100
  + Ignore the test score
  + Display an error message
  + Prompt the user for another score



LB5. Implement LB4 as a GUI program.



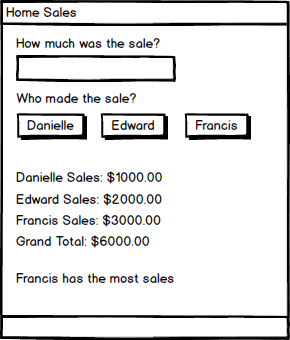
LB6. Write a console application that tracks the home sales of three salespeople (Danielle, Edward, Francis):

* The program prompts the user for a salesperson initial (d, e, or f) and the sales amount, then updates the total sales.
* Any user can have multiple sales.
* If the user enters "z" for the initial then the program should display the total sales for each three salesperson, the grand total for all sales, and the name of the salesperson with the highest total.
* If the user enters an initial other than "d", "e", "f" or "z" then the program should display an error and prompt the user for a new initial.
* All initials should be treated in a case-insensitive manner.

*Program must pass all provided unit tests. (To do so, you will need to follow the format to the right closely.)*

LB7. Implement LB6 as a GUI program.

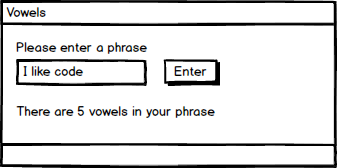
*Program must pass all provided unit tests.*



LB8. Write a GUI application that counts the number of vowels in a phrase that is entered by the user. For this exercise, count both uppercase and lowercase vowels, but do not consider y to be a vowel.

Hint: Research and use String.Substring()

Hint: Use a switch statement for this one.



LB9. Write a console-based program that prints out a multiplication table.

* Prompt the user for the number of rows.
* Prompt the user for the number of columns.
* Display a multiplication table for the numbers 1..rows by 1..cols

