**Constraints:**

1. Use as many descriptive names as possible. I do not want to see myInt1 - 100!
2. For training purposes, most if not all the data for these methods can come from user input.
3. Each one of these methods should be called from Main() and exist in the Program class.
4. Be as creative as you want, the goal is to have working methods but if you want to make them look nice, go for it. As in Num 6 w/ age and voting method, you can say “You may vote” or “Looks like your too young this time!” based on their age.
5. This project should work in a manner where if someone forks/clones this from GitHub, they can just press the play button and have every method run in sequence, so all interface dialogue should be in place already. When finished, push this to GitHub!

**Luke Warm:**

1. Write a method that will print to the console all numbers 1000 through -1000.
2. Write a method that will print to the console numbers 3 through 999 by 3 each time.
3. Write a method to accept two integers as parameters and check whether they are equal or not.
4. Write a method to check whether a given number is even or odd.
5. Write a method to check whether a given number is positive or negative.
6. Write a method to read the age of a candidate and determine whether they can vote. Hint: use Parse()... or the safer TryParse() for an extra challenge!!

**Heatin Up:**

1. Write a method to check if an integer (from the user) is in the range -10 to 10.
2. Write a method to display the multiplication table(from 1 to 12) of a given integer.
3. Write a method that takes a number from the user and returns an array with that many indexes. The indexes shall be random numbers.
4. Write a method to compute the sum of all the elements in an array of integers.
5. Write a method to display the cube of the number up to given an integer. (If a user inputs the number 3 the method should print to the console: Number is: 1 and the cube of 1 is:1, Number is: 2 and the cube of 2 is: 8, Number is: 3 and the cube of 3 is:27)