Exercise 23: Keeping the Odd

Although this exercise isn't worth any points, it gives you valuable programming experience. You're almost definitely going to have to complete the exercises to succeed in the course.

**Getting Started**

Create a new Console Application named Exercise20. Add a list variable to hold integers and create the list object for that variable.

Remember, the **List** class is in the **System.Collections.Generic** namespace, so you may need to add a using directive for that namespace.

**Problem 1 - Populate and print the list**

Use a for loop to populate the list with the integers from 1 to 10 (including 1 and 10).

Use a for loop to print the contents of the list. The **List** class has a **Count** property that tells how many elements the list contains. You should use that property as appropriate in your for loop condition rather than hard-coding how many times the loop body will execute.

**Problem 2 - Use a for loop to remove even numbers from the list**

Use a backward for loop, starting at the last index in the list, to remove the even numbers from the list. You need to look at the value of each element in the list and only remove it if the value of the element is even; we're not removing the elements at even **indexes**, we're removing **elements** that are even numbers. Use a for loop to print the contents of the list.

Note: By ugly coincidence, this would also work with a forward for loop because the elements that "escape processing" would all be odd numbers.

**Problem 3 - Observe bad forward for loop removal**

Populate a list with the numbers from 1 to 5 inclusive. Use a forward for loop (starting i at 0) to remove 1, 2, and 3 from the list. You need to look at the value of each element in the list and only remove it if the value of the element is 1, 2, or 3; we're not removing the elements at **indexes** 1, 2, and 3, we're removing **elements** that have the values 1, 2, or 3.

Print the list to see that the element with the value 2 gets skipped over, staying in the list.