

## Lab 1 – Generics & Linear Data Structure

**Due Date:** Midnight of week3's Friday

**Purpose:** The purpose of this assignment is to help you:

- Become familiar with Linear data structure
- Get hands-on experiences on generics, collections and extension method

**Instructions:** Be sure to read the following general instructions carefully:  
 This assignment should be completed individually by all the students. Submit your solution **through the dropbox**. You must name your submission according to the following rule: **studentID(yourlastname)\_LABnumber.zip**. e.g.,  
 300123456(smith)\_Lab#1.zip

### Rubric

	<b><u>Functionality</u></b>	<b><u>Marks</u></b>
Q1	2.1 Implementation of the extension methods	1.5*2
	2.2 Consume the extension method	1*2
Q2	3.1 Finish class <code>SinglyLinkedList&lt;E&gt;</code>	10+4
	3.2 Load data from <code>Auto_mpg.csv</code> , and insert them into a Singly Linked list, you can either add vehicle info at the first or at the end	
	3.2.1 Modelling the data in <code>Auto_mpg.csv</code> so that each piece of information can be handled separately	2
	3.2.2 load data from csv file	2
	3.2.3 insert vehicles' info to a <i><b>SinglyLinkedList</b></i>	1
	3.2.4 print out all vehicles on the screen in a good format	1
Overall	Overall (code quality, application usability, etc.)	1

### Question 1 [4 marks]

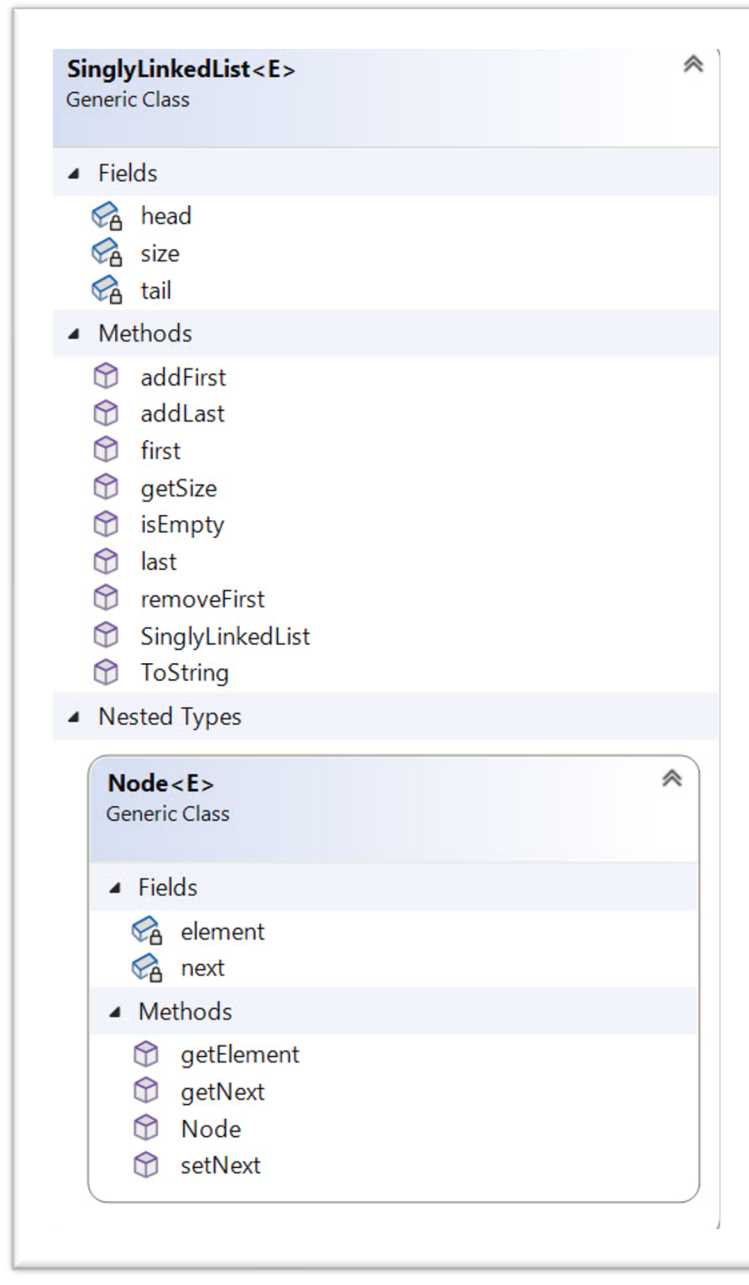
***ObservableCollection<T>*** class represents a dynamic data collection that provides notifications when items get added or removed, or when the whole list is refreshed.

You are asked to implement 2 extension methods for ***ObservableCollection<T>*** class to

1. Add all elements in a ***List<T>*** to the end of the ***ObservableCollection<T>*** object.
2. Remove all occurrences of items in a ***List<T>*** from an ***ObservableCollection<T>*** object
3. Test the implemented extension methods.

### Question 2 [20 marks]

Implement generic class ***SinglyLinkedList<E>***, including its nested class ***Node<E>***



Implement a C# application to consume the implemented ***SinglyLinkedList<E>***. More specifically, load the data from ***Auto\_mpg.csv*** and store those data into a singly linked list.

1. Insert data line by line, the insertion can be occurred at the head of your singly linked list, ***or*** at the end of your singly linked list
2. print out all elements in the singly linked list in a good format (e.g., one line per vehicle).