22F CST 8284 Object Oriented Programming (JAVA)

ASSIGNMENT 3

SUBMISSION INFORMATION

1. You must submit your solutions on Brightspace or before the due date:
   Wednesday, **December 7 at 11:59PM.**
2. You are required to provide all your solutions in CarBrandsList.java file which you must submit. Do not create another class.
3. Generate and submit Javadoc file showing the comments that are in your code.
4. Submit in your designated BrightSpace area for your Lab section:
5. If your professor provides an alternative submission instruction, then name your files for submission according to the submission instructions provided by your lab Professor.
6. Name your submission using the convention (marks could be deducted if not used):
   FirstName_LastName_LabSectXXX_Assignment3.

Description

A collection is referred to as a data structure (object) that can hold references to other objects. Collections contain references to object types which has an *is-a-relationship.* Its framework interface states the operations to be performed generically on different types of collections.

An advantage of collection is that it provides API consistency such that the classes that implement its set of interfaces such as Set, List or Map all have some common set of methods.

In this assignment, you are given a lists of car brands and required to perform some operations. Among other things, you will explore some interesting collection methods and concepts such as Lists, LinkedLists, ListIterators and more. Some specific methods have been mentioned to you, which you must incorporate into your work to get the marks.

The content of the theory slide deck from this week's lecture provides some resource for completing your work successfully.

Files Provided for You

a. The solution file named **CarBrandsList.java** that contains the main method.
b. Two arrays of car brands:

> ➢ *rides* containing: *cardillac, toyota, suzuki, chevrolet, hyundai and mercedies.*
> ➢ *rides2* containing: *volvo, subaru, volkswagen, nissan, cardillac, toyota and honda.*

   c. The sample output file – just a guide to your implementation.

Your Tasks

In this assignment you are required to:

1. Note that no import statements have been included in your starter code. You are required to provide them by yourself.
2. Create two LinkedLists of Strings namely list1 and list2.
3. Add the content of *rides* to list1 and then *rides2* to list2.
4. In order to make your code more concise, use the asList method and the LinkedList constructor that takes a Collection argument.
5. Add the elements in list*2* to list1 using the addAll method.
6. Create a method named printList() and use it to print out the updated content of list1 and then release list2 resource.
7. Using a suitable method, convert all the elements in list1 to uppercase and print out this update.
8. Going forward, certain car brands in the list are no longer needed. Using the appropriate method (obtain the sublist and use the clear method to delete sublist items), delete car brands (in positions) 5 to 7 from the list, and then print out the updated output at this point.
9. Now, print the current list in reverse order using the printReversedList method.
10. Using the sort method, sort the list to ensure that the contents are in alphabetical order. Print the list to show this expected order.
11. Having observed that some car brands have duplicates, use the printNonDuplicates method to eliminate all duplicated elements and then print out the final content of the list.

Hint: All methods are called from main – check the order of the output.

RUBRICS (8%)

Item	Neglected (%)	Partially completed (%)	Completed correctly including output (%)	Possible Mark deduction (%)
Submission with compliance to instruction	0		2	Incorrect
Javadoc Style documentation of code	0		15	Incorrect
Array and LinkedList implementation	0		10	Incorrect
Correct creation and use of the specified methods	0		35	Incorrect
Correct use of print methods and output display	0		10	Incorrect
Code execution	0		8	Code completed but not running and/or incorrect output

Further marks can be deducted if general coding / submission criteria as have been used in this course are not followed. Ask for clarity where needed.