HashTable Lab
(Used Cars Database)
Advanced Computer Science

Topics: Objects, Classes, Java Components, HashTable, Linked List

Lab Description:
In this lab, you will create a database of used cars that are available for sale.

Car
Each car in the database will contain the make, model name, year, and price.
The make is the manufacturer of the car. The hashCode generated by each car will be ordered by the first 3 letters (or less) of the make. You must come up with your own algorithm for generating the hashcode. Do not use String's hashCode method.
Your program will have at least 10 cars to begin with. At least 3 cars have the same make. Another set of 3 cars have the same make that is different.
(When testing this lab, we will assume that at most you'll have 5 cars for each make.)

Database Structure
The cars will be stored in **Hashtable** type database. Each bucket will be chained with cars of the same make. The hashCode generated by each car will be ordered by the first 3 letters (or less) of the make. For example, Toyota will use "toy" to generate the hashcode, and Tesla will use "tes" to generate the hashcode. The hashTable will be somewhat sorted alphabetically if you go through each bucket.

List Selection Menu
A list menu will display all available car makes. (See the resource page for an example of a list menu.) When user the clicks on the car make, it will display all the available cars. When displaying each car, you will display the car make, model name, year, and price. (You will have buttons next to each car depending on the view. See below.)

Dealer View
The dealer can view the entire inventory using a List Selection Menu.
The dealer can add new cars to the inventory.
For each car displayed, there will be a remove and update button for dealer to remove the car or update the car's price.

Consumer View
The consumer can view the entire inventory using a List Selection Menu.
For each car displayed, there will be a buy button for the consumer to buy(remove). This will remove the car from the inventory.

Car Class
This stores the information of each car.

HashTable<E> Class
This is where you will manage the grid of Article objects.

DLList<E> Class
This will be used in your HashTable<E> class.

Node<E> class
This will be used in your DLList<E> class.

You cannot use any other data structures (e.g. ArrayList, LinkedList, etc…) than the ones above.

Grading Rubric:
Your lab will be graded using the following criteria.

Completion of Lab as Described	90
Milestone 1 - **Due ~~Monday~~ Tuesday.** Have a list of 5 cars. 3 of the cars must be from the same make. When you click on the make of the car, all the cars from that make will be displayed.	10

You will have class time to work on this lab. Any late submission or revision will be given a maximum score of 90.

Due: See Google Classroom. Link

Suggestions for converting strings to HashCode.

- To convert each char to a number, you can cast it.

```
char myChar = 'a';
int value = (int) myChar; //'a' returns 97, for 'a' to be 1, subtract 96.
```

Think of the following

```
a = 1       aa = 27     ba = 53
b = 2       ab = 28     bb = 54
c = 3       ac = 29     ...
...         ...
z = 26            az = 52
a = a(26^0) = 1(1) = 1
z = z(26^0) = 26(1) = 26
aa = a(26^1) + a(26^0) = 1(26) + 1(1) = 27
ab = a(26^1) + b(26^0) = 1(26) + 2(1) = 28
ba = b(26^1) + a(26^0) = 2(26) + 1(1) = 53
ba = b(26^1) + b(26^0) = 2(26) + 2(1) = 54
```