Assignment Prefix: Lab07

Due Date: Friday, Oct. 14th @ 11:59pm

Points: 100

This is an individual assignment.

Restrictions: you cannot use any methods from the Java Array(s) class to copy an array, check for equality, or otherwise manipulate an array. You must write the Java code to perform these functions.

The above restrictions do not apply to any of the classes copied from the textbook.

Create a NetBeans project named Lab07 and ensure it is saved to a location like desktop or your flash drive. In the project you will do the following:

For this assignment you are going to explore the wonderful world of iterators.

Step 1:

Create a class named **LuckyNumber** that has the following instance variables:

- String name
- int luckyNumber

This class should **only have** a single overload constructor that:

- takes a single parameter, name
- automatically assigns luckyNumber a random number between 0 and 9 (inclusive of both) to the instance being created

This class **must have** the following methods:

- getName
- get luckyNumber
- toString
    - make sure your toString returns the class name
    - the class name will be useful if you "become lost" in your data structure
- equals

This class **should not have** the following methods:

- set name
    - you are who you are
- set luckyNumber
    - you can't change your luck

Step 2:

Create a class named **LuckyNumberList** that will contain a LinkedPositionalList of LuckyNumber objects.

This class should contain a constructor and methods similar to the Alphabet class in the IteratorExample1 project.

- Unlike the Alphabet class your constructor should just create an empty LinkedPositionalList
- You will need to add an **addLuckyNumber** method to this class
    - This method has a single parameter of type LuckyNumber.
    - this method should add the new entry at the end of the list.
- It will be the job of the client class the add entries into the list.

You will be adding two custom iterators:

- an iterator that iterates over the positions in the list that have an even LuckyNumber
- an iterator that iterates over the positions in the list that have a prime LuckyNumber
- Note, if you use the Alphabet class as a guide you will remove the VowelPositionIterator. Actually, you may find it helpful to modify the VowelPositionIterator into one of the above required iterators.

Step 3:

Create a client class named **Lab07Client** that does the following:

- Creates an instance of your LuckyNumberList class.
- Fills the list with at least ten (10) names.
    - You can hardwire the fills into your code by using ten calls to addLuckyNumber that pass new instances of LuckyNumber objects.
    - Names can be hardwired into your code.
- Prints out the resulting contents of your instance of the LuckyNumberList class using:
    - The default iterator that displays all entries in the list
    - The Prime iterator that displays only the entries in the list that have prime luckyNumbers
    - The Even iterator that display only the entries in the list that have even luckyNumbers.