Linked Lists

In this assignment you will write a program that implements a variant of a linked list. This variant has a dummy node pointed to by the head link as shown in the following figure:

![Linked list with a dummy first node](image)

This trick will allow your code to be a little simpler, not requiring a special case for add or remove operations at index 0. Your constructor method will be:

```java
public LinkedList(){
    head = new Node(null);
    size = 0;
}
```

You need to write a class called `LinkedList` that implements the following `List` interface:

```java
// a list interface
public interface List {
    public boolean isEmpty();  // returns true if the list is empty, false otherwise
    public int size();  // returns the number of items in the list
    public void add(Object item);  // adds an item to the list
}
```
public void add(int index, Object item);
// adds an item to the list at the given index
// precondition: none
// postcondition: item is added at the given index;
// the indices of following items are increased by 1.

public void remove(int index);
// removes the item from the list that has the given index
// precondition: none
// postcondition: removes the first item in the list whose equal method
// matches that of the given item

public void remove(Object item);
// removes an item from the list
// precondition: none
// postcondition: removes the first item in the list whose equal method
// matches that of the given item; the indices of the following items are
// decreased by 1

public List duplicate();
// creates a duplicate of the list
// precondition: none
// postcondition: returns a copy of the linked list

public List duplicateReversed();
// creates a duplicate of the list with the nodes in reverse order
// precondition: none
// postcondition: returns a copy of the linked list with the nodes in
// reverse order

In addition to the interface, your LinkedList class needs to implement a toString() method that prints the list in the format

[ size: the_size_of_the_list item1, item2, .... ]

Specifications, notes, and hints
Your program needs to meet the following specifications:
- Submit the file LinkedList.java. Your Node class should be an inner class within the LinkedList class. Make sure your class implements the interface as specified, i.e. your class should begin with public class LinkedList implements List.

- None of your methods should contain a test for the index being equal to 0, as the point is to have a simpler implementation.

- When commenting your code use Javadoc style comments at the beginning of each method.

- Put comments at the top of the file with your name, EID, email address, date and course, and a short (one or two line) description of what the program does. We will be testing the code on the machines in the CS computer lab, so make sure your code runs on those machines.

- Submit your source code files via the checkin program by the due date (read the course syllabus for the late policy).