***LINKED LISTS Name\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***You may reference the following code in your problems. For each problem you may assume there is at least one node in the list***

public class LinkedList

{

private class ListNode

{

public Object data;

public ListNode next;

private ListNode(Object da, ListNode ne){

data = da;

next = ne;

}

}

private ListNode first; // points to the first element, null if empty

private int size; // the number of elements in the list

public LinkedList()

{

first = null;

}

}

//method removeFirst will remove the First Node in the Linked List

public void removeFirst()

{

}

Complete method addToEnd(Object value).

//method addToEnd will install a new Node containing the parameter value

// at the end of the LinkedList

public void addToEnd(Object value)

{

}

Complete method removeAllExceptLast()

//method removeAllExceptLast will result in the LinkedList containing

//only one Node – the one originally at the end of the LinkedList

public void removeAllExceptLast()

{

}