```
/**
 * Implementation of a Singly-Linked List.
 *
 * @author  __your_name___
 */

import java.util.NoSuchElementException;

public class SLinkedList<E>
{
    // Representation of the list nodes
    private class Node
    {
        E data;          // the data value stored at the node
        Node next;       // the successor of this node

        // creates a node with the given data item and no successor
        Node(E d)
        {
            data = d;
            next = null;
        }
    }

    /**
     * The first node in the list.
     */
    private Node head;

    // put comment in Javadoc style
    public SLinkedList()
    {
        head = null;
    }

    // put comment in Javadoc style
    public boolean isEmpty()
    {
        return head == null;
    }

    /**
     * Adds the given element to the front of the list.
     *
     * @param data    the element to add
     */
    public void addFirst(E data)
    {
        Node node = new Node(data);
        node.next = head;
        head = node;
    }

    /**
     * Returns the first element in the list.
     *
     * @return the first element in the list
     * @throws NoSuchElementException when the list is empty
     */
    public E getFirst()
    {
        if (this.isEmpty()) {
            throw new NoSuchElementException();
        }

        return head.data;
    }

    // put comment in Javadoc style
    public void addLast(E data)
    {
        // special case
        if (this.isEmpty()) {
            head = new Node(data);
        }
        else {
            // find last node
            Node curr = head;
            while (curr.next != null) {
                curr = curr.next;
            }

            // attach the new node to the last node
            Node node = new Node(data);
            curr.next = node;
        }
    }

    // put comment in Javadoc style
    public String toStringNext()
    {
        String str = "";
        Node curr = head;

        // add each data item to the result string
        while (curr != null) {
```

```java
            str = str + curr.data + " ";
            curr = curr.next;
        }

        // remove trailing space and enclose in [ ]
        str = "[" + str.trim() + "]";

        return str;
    }
}
```