

```
2 * Implementation of a Singly-Linked List.
6
7 import java.util.NoSuchElementException;
8
9 public class SLinkedList<E>
10 {
11     // Representation of the list nodes
12     private class Node
13     {
14         E data;           // the data value stored at the node
15         Node next;       // the successor of this node
16
17         // creates a node with the given data item and no successor
18         Node(E d)
19         {
20             data = d;
21             next = null;
22         }
23     }
24
25     /**
26      * The first node in the list.
27      */
28     private Node head;
29
30     // put comment in Javadoc style
31     public SLinkedList()
32     {
33         head = null;
34     }
35
36     // put comment in Javadoc style
37     public boolean isEmpty()
38     {
39         return head == null;
40     }
41
42     /**
43      * Adds the given element to the front of the list.
44      *
45      * @param data    the element to add
46      */
47     public void addFirst(E data)
48     {
49         Node node = new Node(data);
50         node.next = head;
51         head = node;
52     }
53 }
```

```
53
54  /**
55   * Returns the first element in the list.
56   *
57   * @return the first element in the list
58   * @throws NoSuchElementException when the list is empty
59   */
60  public E getFirst()
61  {
62      if (this.isEmpty()) {
63          throw new NoSuchElementException();
64      }
65
66      return head.data;
67  }
68
69  // put comment in Javadoc style
70  public void addLast(E data)
71  {
72      // special case
73      if (this.isEmpty()) {
74          this.addFirst(data);
75      }
76      else {
77          // find last node
78          Node curr = head;
79          while (curr.next != null) {
80              curr = curr.next;
81          }
82
83          // attach the new node to the last node
84          Node node = new Node(data);
85          curr.next = node;
86      }
87  }
88
89  // put comment in Javadoc style
90  public String toStringNext()
91  {
92      String str = "";
93      Node curr = head;
94
95      // add each data item to the result string
96      while (curr != null) {
97          str = str + curr.data + " ";
98          curr = curr.next;
99      }
100  }
```

```
101 // remove trailing space and enclose in [ ]
102 str = "[" + str.trim() + "];
103
104 return str;
105 }
106 }
```