```java
/*************************************************************
* Dalton Nofs                                                *
* Login ID: nofs5491                                         *
* CS-102, Summer 2017                                        *
* Programming Assignment 5                                   *
* Node class:  node object for linkedLists                     *
*************************************************************/
public class Node<T>
{
    T data;    // Data to be stored in Node
    Node<T> next;        // Next node ptr
    Node<T> previous;    // Previous node ptr

    /*************************************************************
    * Method: Node()                                             *
    * Purpose: default constructor for node obj                    *
    *                                                              *
    * Parameters:               N/A                              *
    * Returns: void:            N/A                              *
    *************************************************************/
    public Node()
    {
        data = null;
        next = null;
        previous = null;
    }

    /*************************************************************
    * Method: getData()                                            *
    * Purpose: get previous node or next node                     *
    *                                                              *
    * Parameters:               N/A                              *
    * Returns: Object:          data type stored in node          *
    *************************************************************/
    public T getData()
    {
        return data;
    }

    /*************************************************************
    * Method: setData()                                            *
    * Purpose: get previous node or next node                     *
    *                                                              *
    * Parameters: T:            Data to be stored in node        *
    * Returns: void:            N/A                              *
    *************************************************************/
    public void setData(T data)
    {
        this.data = data;
    }

    /*************************************************************
    * Method: getNext()/getPrevious()                            *
    * Purpose: get previous node or next node                     *
    *                                                              *
    * Parameters:               N/A                              *
    * Returns: Node:            Node pointing to next or prev    *
    *************************************************************/
    public Node<T> getNext()
    {
        return next;
    }
    public Node<T> getPrevious()
    {
        return previous;
    }

```

```
68      /*************************************************************
69      * Method: setNext()/setPrevious()                            *
70      * Purpose: Set the next or previous node                     *
71      *                                                               *
72      * Parameters: Node:          Next/previous node              *
73      * Returns: Void:             N/A                             *
74      *************************************************************/
75      public void setNext(Node<T> next)
76      {
77          this.next = next;
78      }
79      public void setPrevious(Node<T> previous)
80      {
81          this.previous = previous;
82      }
83
84  }
```