Linked List in java

In Java, LinkedList class implements the list interface.

This class consists of the following methods :

1. **boolean add(Object element)** : It appends the element to the end of the list.

2. **void add(int index, Object element)**: It inserts the element at the position 'index' in the list.

3. **void addFirst(Object element)** : It inserts the element at the beginning of the list.

4. **void addLast(Object element)** : It appends the element at the end of the list.

5. **boolean contains(Object element)** : It returns true if the element is present in the list.

6. **Object get(int index)** : It returns the element at the position 'index' in the list. It throws 'IndexOutOfBoundsException' if the index is out of range of the list.

7. **int indexOf(Object element)** : If element is found, it returns the index of the first occurrence of the element. Else, it returns -1.

8. **Object remove(int index)** : It removes the element at the position 'index' in this list. It throws 'NoSuchElementException' if the list is empty.

9. **int size()** : It returns the number of elements in this list.

10. **void clear()** : It removes all of the elements from the list.

```
// Java code for Linked List implementation

import java.util.*;

public class Test
{
    public static void main(String args[])
    {
        // Creating object of class linked list
        LinkedList<String> object = new LinkedList<String>();

        // Adding elements to the linked list
        object.add("A");
        object.add("B");
        object.addLast("C");
        object.addFirst("D");
        object.add(2, "E");
        object.add("F");
        object.add("G");
        System.out.println("Linked list : " + object);

        // Removing elements from the linked list
        object.remove("B");
        object.remove(3);
        object.removeFirst();
        object.removeLast();
        System.out.println("Linked list after deletion: " + object);

        // Finding elements in the linked list
        boolean status = object.contains("E");

        if(status)
            System.out.println("List contains the element 'E' ");
        else
            System.out.println("List doesn't contain the element 'E'");

        // Number of elements in the linked list
        int size = object.size();
        System.out.println("Size of linked list = " + size);

        // Get and set elements from linked list
        Object element = object.get(2);
        System.out.println("Element returned by get() : " + element);
        object.set(2, "Y");
        System.out.println("Linked list after change : " + object);
    }
}
```

Output :

```
Linked list : [D, A, E, B, C, F, G]
Linked list after deletion: [A, E, F]
List contains the element 'E'
Size of linked list = 3
Element returned by get() : F
Linked list after change : [A, E, Y]
```