```
class Node<S> {
    S data;
    Node<S> next;

    public Node(S data) {
        this.data = data;
        this.next = null;
    }
}

class SinglyLinkedList<S> {
    private Node<S> head;

    public SinglyLinkedList() {
        this.head = null;
    }

    public void addFirst(S s) {
        Node<S> newNode = new Node<>(s);
        newNode.next = head;
        head = newNode;
    }

    public void rotate() {
        if (head == null || head.next == null) {
            return;
        }

        Node<S> current = head;
```

```
while (current.next != null && current.next.next != null) {
                current = current.next;
                {

        Node<S> secondToLast = current;
        Node<S> last = current.next;

        secondToLast.next = null;
        last.next = head;
        head = last;
        {

    public void printList() {
    Node<S> current = head;
    while (current != null) {
System.out.print(current.data + " ");
        current = current.next;
        {
    System.out.println();
        {
    {

    public class Main {
public static void main(String[] args) {
SinglyLinkedList<Integer> list = new SinglyLinkedList<>();
        list.addFirst(10);
        list.addFirst(20);
        list.addFirst(30);
        list.addFirst(40);
```

```
System.out.print("Before rotate: ");
list.printList();

list.rotate();

System.out.print("After rotate: ");
list.printList();
{
  {
```