6. Write a program in Java to insert a new element in a sorted circular linked list

```java
class Node {
    int data;
    Node next;

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


class CircularLinkedList {
    Node head;

    public CircularLinkedList() {
        head = null;
    }

    public void insert(int data) {
        Node newNode = new Node(data);

        if (head == null) {
            head = newNode;
            head.next = head;
        } else if (data <= head.data) {
            Node last = getLastNode();
            newNode.next = head;
            last.next = newNode;
            head = newNode;
        } else {
            Node current = head;
```

```java
        while (current.next != head && data > current.next.data) {

            current = current.next;

        }

        newNode.next = current.next;

        current.next = newNode;

    }

}


public Node getLastNode() {

    Node temp = head;

    while (temp.next != head) {

        temp = temp.next;

    }

    return temp;

}


public void display() {

    if (head == null) {

        System.out.println("The circular linked list is empty.");

        return;

    }


    Node current = head;

    do {

        System.out.print(current.data + " ");

        current = current.next;

    } while (current != head);

    System.out.println();

}
}
```

```java
public class Main {

    public static void main(String[] args) {

        CircularLinkedList list = new CircularLinkedList();

        // Inserting elements into the sorted circular linked list
        list.insert(10);

        list.insert(20);

        list.insert(30);

        list.insert(40);

        System.out.println("Original list:");

        list.display();

        // Inserting a new element
        list.insert(25);

        System.out.println("List after inserting 25:");

        list.display();
    }
}
```

OUTPUT:

Original list:

10 20 30 40

List after inserting 25:

10 20 25 30 40