

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

```
package com.assin.solution;
```

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

```
public class SortedCirList {
```

```
    public static Node insert(Node head, int data) {  
        Node newNode = new Node(data);
```

```
        if (head == null) {  
            newNode.next = newNode;  
            return newNode;  
        }
```

```
        Node current = head;
```

```
        // If the new node is to be inserted before the head
```

```
        if (data < head.data) {  
            while (current.next != head) {  
                current = current.next;  
            }  
            current.next = newNode;  
            newNode.next = head;  
            return newNode;  
        }
```

```
        // Find the node after which the new node is to be inserted
```

```
        while (current.next != head && current.next.data < data) {  
            current = current.next;  
        }
```

```
        newNode.next = current.next;  
        current.next = newNode;  
        return head;  
    }
```

```
    public static void displayCircularLinkedList(Node head) {  
        if (head == null) return;
```

```
        Node current = head;
```

```

    do {
        System.out.print(current.data + " ");
        current = current.next;
    } while (current != head);
    System.out.println();
}

public static void main(String[] args) {
    Node head = new Node(1);
    head.next = new Node(3);
    head.next.next = new Node(5);
    head.next.next.next = head;

    System.out.println("Circular Linked List before insertion:");
    displayCircularLinkedList(head);

    int newData = 4;
    head = insert(head, newData);

    System.out.println("Circular Linked List after inserting " + newData + ":");
    displayCircularLinkedList(head);
}
}

```