Name: Ch.Navyasri

Reg no :23bce9703

Write a dsa program to merge two single linked list

```
class Node {
  int data;
  Node next;
    Node(int data) {
    this.data = data;
    this.next = null;
  }
}
class LinkedList {
  Node head;
  public void insert(int data) {
    Node newNode = new Node(data);
    if (head == null) {
      head = newNode;
```

```
} else {
    Node temp = head;
    while (temp.next != null) {
      temp = temp.next;
    }
    temp.next = newNode;
  }
}
 public Node mergeLists(Node I1, Node I2) {
  if (I1 == null) return I2;
  if (I2 == null) return I1;
       Node dummy = new Node(0);
  Node current = dummy;
  while (I1 != null && I2 != null) {
    if (l1.data <= l2.data) {
      current.next = I1;
      I1 = I1.next;
    } else {
      current.next = I2;
      12 = 12.next;
    }
    current = current.next;
```

```
}
    if (l1 != null) {
      current.next = I1;
    } else if (l2 != null) {
      current.next = I2;
    }
    return dummy.next;
    public void printList(Node head) {
    Node temp = head;
    while (temp != null) {
      System.out.print(temp.data + " ");
      temp = temp.next;
    }
    System.out.println();
  }
public class Main {
  public static void main(String[] args) {
    LinkedList list1 = new LinkedList();
    LinkedList list2 = new LinkedList();
```

}

```
list1.insert(1);
    list1.insert(3);
    list1.insert(5);
        list2.insert(2);
    list2.insert(4);
    list2.insert(6);
         LinkedList mergedList = new LinkedList();
     mergedList.head = mergedList.mergeLists(list1.head, list2.head);
        System.out.println("Merged List:");
    mergedList.printList(mergedList.head);
  }
}
```

Output:

```
C:\>cd C:\Users\NAVYA SRI\Desktop\23BCE9703
C:\Users\NAVYA SRI\Desktop\23BCE9703>set path = "C:\Program Files\Java\jdk-22\bin"
C:\Users\NAVYA SRI\Desktop\23BCE9703>javac Main.java
C:\Users\NAVYA SRI\Desktop\23BCE9703>java Main
Merged List:
1 2 3 4 5 6
```