

5. Write a program in Java to delete the first occurrence of a key in a singly linked list class Node

```
int data;
Node next;

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

class LinkedList {
    Node head;

    void deleteKey(int key) {
        Node current = head;
        Node previous = null;

        // If the key is found at the head node
        if (current != null && current.data == key) {
            head = current.next;
            return;
        }

        // Traverse the list until the key is found or the end is reached
        while (current != null && current.data != key) {
            previous = current;
            current = current.next;
        }

        // If the key is found, remove the node
```

```
    if (current != null) {  
        previous.next = current.next;  
    }  
}
```

```
void insert(int data) {  
    Node newNode = new Node(data);  
  
    if (head == null) {  
        head = newNode;  
    } else {  
        Node current = head;  
        while (current.next != null) {  
            current = current.next;  
        }  
        current.next = newNode;  
    }  
}
```

```
void display() {  
    Node 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) {
```

```
LinkedList list = new LinkedList();

// Inserting elements into the linked list
list.insert(10);
list.insert(20);
list.insert(30);
list.insert(40);
list.insert(50);

System.out.println("Linked list before deletion:");
list.display();

// Deleting the first occurrence of a key
int key = 30;
list.deleteKey(key);

System.out.println("Linked list after deleting first occurrence of " + key + ":");
list.display();
}
}
```

OUTPUT:

Linked list before deletion:

10 20 30 40 50

Linked list after deleting first occurrence of 30:

10 20 40 50