

Name: Yash Daga

Registration Number: 20BCE7323

Write a Program to create a Singly Linked List with n nodes and then perform following operations. a) Display the list b) Find Maximum element from SLL c) Find sum elements in SL

Program code:

```
import java.util.*;

public class CountNodes {

    class Node{

        int data;

        Node next;

        public Node(int data) {

            this.data = data;

            this.next = null;

        }

    }

    public Node addNode(int data, Node head) {

        Node newNode = new Node(data);

        Node temp = head;

        if(head == null) {

            head = newNode;

            return head;

        }

        while(temp.next != null) {

            temp = temp.next;

        }

        temp.next = newNode;

        return head;

    }

}
```

```

public int countNodes(Node head) {
    int count = 0;
    Node current = head;

    while(current != null) {
        count++;
        current = current.next;
    }
    return count;
}

public void display(Node head) {
    Node current = head;
    if(head == null) {
        System.out.println("List is empty");
        return;
    }
    System.out.println("Nodes of singly linked list: ");
    while(current != null) {
        System.out.print(current.data + " ");
        current = current.next;
    }
    System.out.println();
}

    public static int largestElement(Node head)
    {
        int max = Integer.MIN_VALUE;
        Node current = head;
        if(head == null) {
            return -1;
        }
        while (current != null)

```

```

        {
            if (max < current.data)
                max = current.data;
            current = current.next;
        }
        return max;
    }

    public static int sumOfNodes(Node head)
    {
        int sum = 0;
        if (head == null)
            return 0;
        Node current = head;
        while(current != null) {
            sum += current.data;
            current = current.next;
        }
        return sum;
    }
}

```

```

public static void main(String[] args) {

```

```

    CountNodes sList = new CountNodes();

```

```

    Scanner sc = new Scanner(System.in);

```

```

        System.out.println("Enter the number of nodes");

```

```

        int n= sc.nextInt();

```

```

        Node head = null;

```

```

        for(int i=1;i<=n;i++)

```

```

        {

```

```

            head = sList.addNode(i, head);

```

```

        }
    }
}

```

```
sList.display(head);  
System.out.println("Count of nodes present in the list: " + sList.countNodes(head));  
    System.out.print("Maximum element in linked list: ");  
System.out.println(largestElement(head));  
    System.out.println( "Sum of nodes = " + sumOfNodes(head));  
}  
}
```

```
Command Prompt
D:\20BCE7323>javac CountNodes.java
D:\20BCE7323>java CountNodes
Enter the number of nodes
5
Nodes of singly linked list:
1 2 3 4 5
Count of nodes present in the list: 5
Maximum element in linked list: 5
Sum of nodes = 15
D:\20BCE7323>
```

```
D:\20BCE7323\CountNodes.java - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Reverse.java Arraysortinsert.java CountNodes.java
1 import java.util.*;
2 public class CountNodes {
3     class Node{
4         int data;
5         Node next;
6
7         public Node(int data) {
8             this.data = data;
9             this.next = null;
10        }
11    }
12
13    public Node addNode(int data, Node head) {
14        Node newNode = new Node(data);
15        Node temp = head;
16        if(head == null) {
17            head = newNode;
18            return head;
19        }
20        while(temp.next != null) {
21            temp = temp.next;
22        }
23        temp.next = newNode;
24        return head;
25    }
26
27    public int countNodes(Node head) {
28        int count = 0;
29        Node current = head;
30
31        while(current != null) {
32            count++;
33            current = current.next;
34        }
35        return count;
36    }
37
38    public void display(Node head) {
39        Node current = head;
40        if(head == null) {
41            System.out.println("List is empty");
42            return;
43        }
44    }
45
46    }
47
48    }
49
50    }
51
52    }
53
54    }
55
56    }
57
58    }
59
60    }
61
62    }
63
64    }
65
66    }
67
68    }
69
70    }
71
72    }
73
74    }
75
76    }
77
78    }
79
80    }
81
82    }
83
84    }
85
86    }
87
88    }
89
90    }
91
92    }
93
94    }
95
96    }
97
98    }
99
100   }
```

length: 2442 lines: 95 Ln: 84 Col: 24 Pos: 2,072 Windows (CR LF) UTF-8 INS

```
Command Prompt
D:\20BCE7323>javac CountNodes.java
D:\20BCE7323>java CountNodes
Enter the number of nodes
5
Nodes of singly linked list:
1 2 3 4 5
Count of nodes present in the list: 5
Maximum element in linked list: 5
Sum of nodes = 15
D:\20BCE7323>
```