Name: Yash Daga

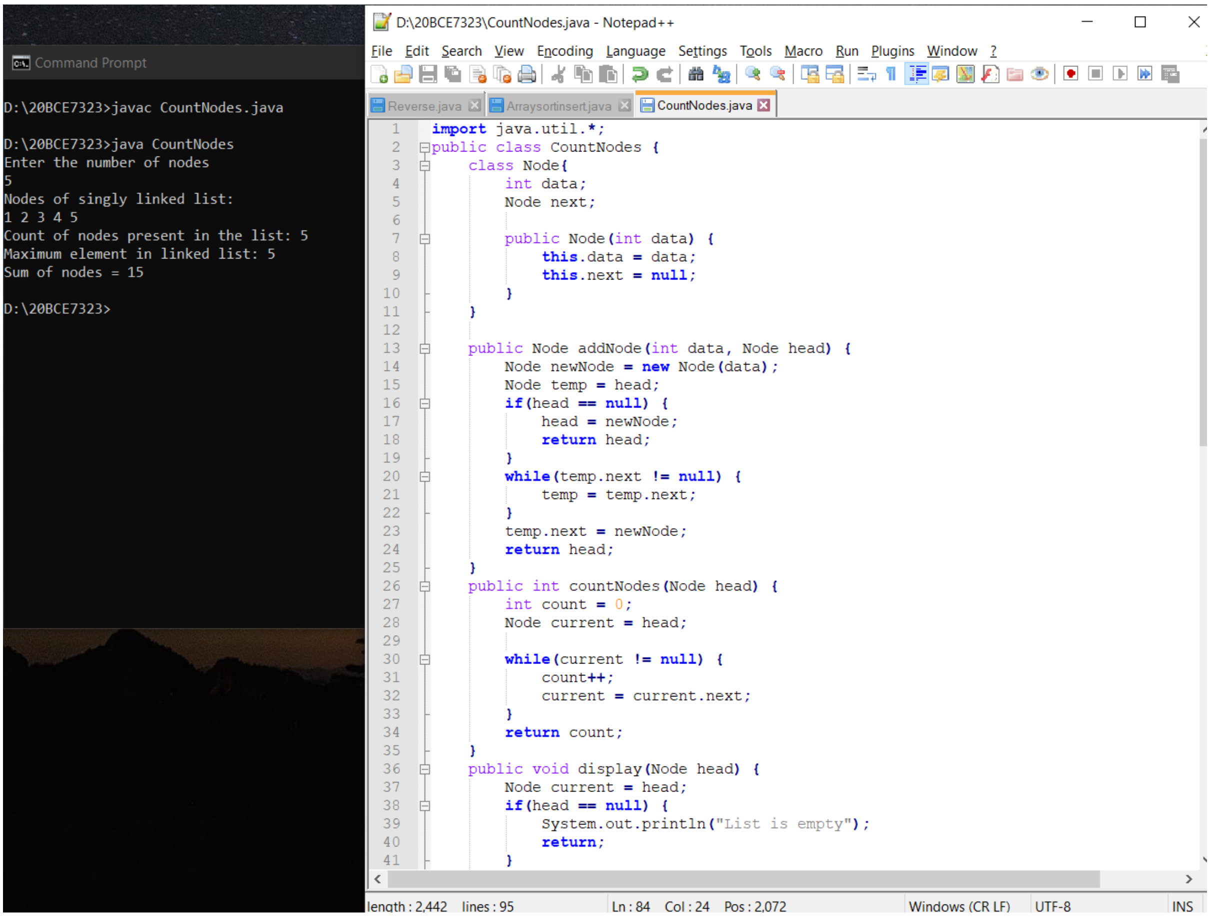
Registration Number: 20BCE7323

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:

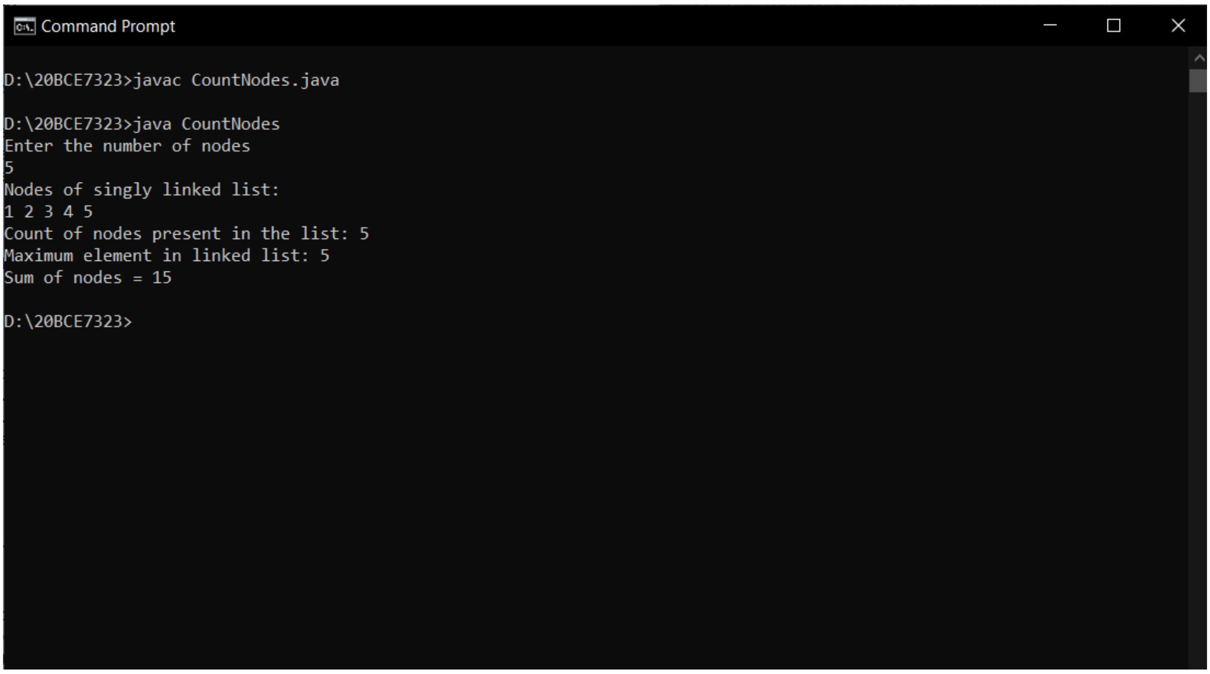
```
import java.util.*;

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\CountNodes.java - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Reverse.java
Arraysortinsert.java
CountNodes.java
length : 2,442 lines : 95
Ln : 84 Col : 24 Pos : 2,072
Windows (CR LF)
UTF-8
INS

```
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>
```

```java
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;
        }
```



```
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>
```