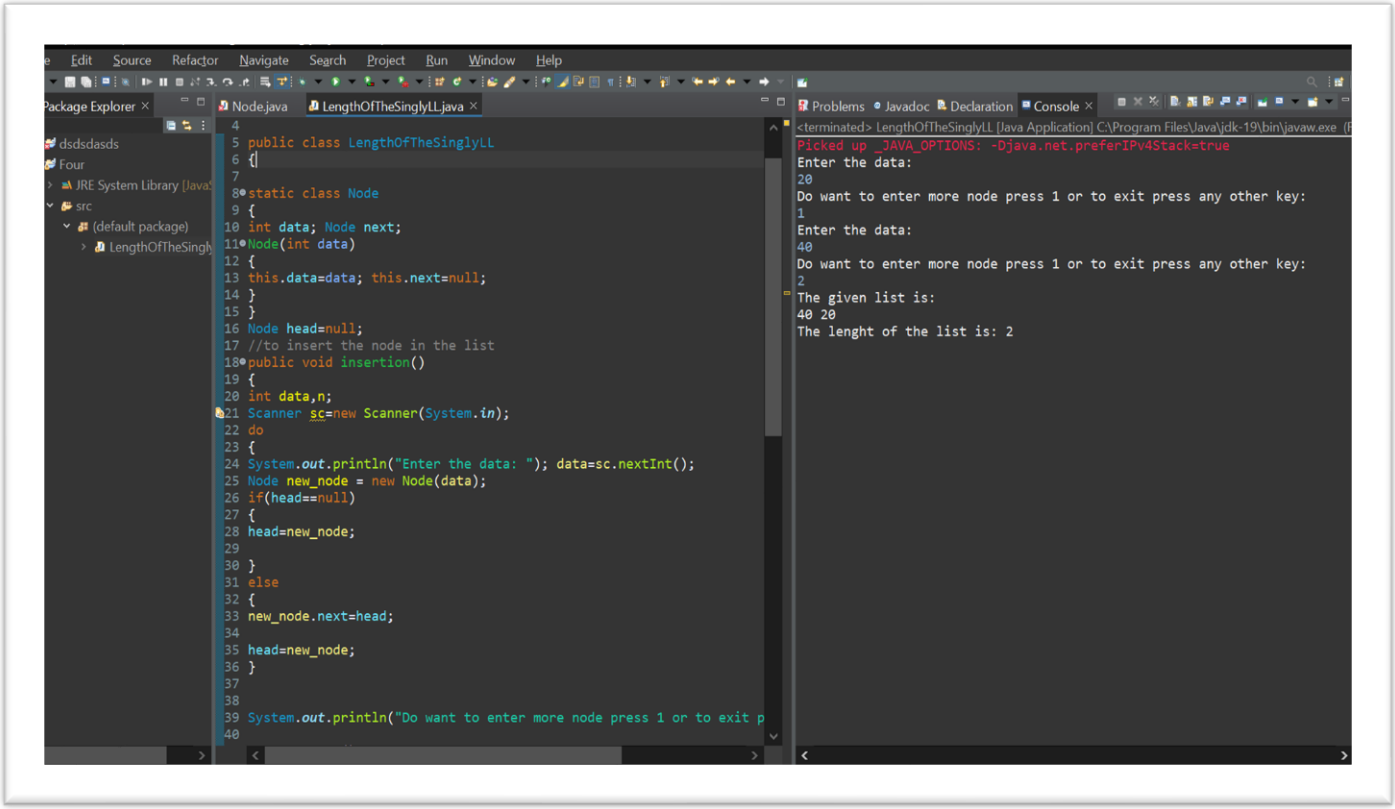
LAB SHEET -4

AIM 1: Understanding the concepts Linked List (10points)

1. **Implement a program to count the length of a singly linked list.**



Code:

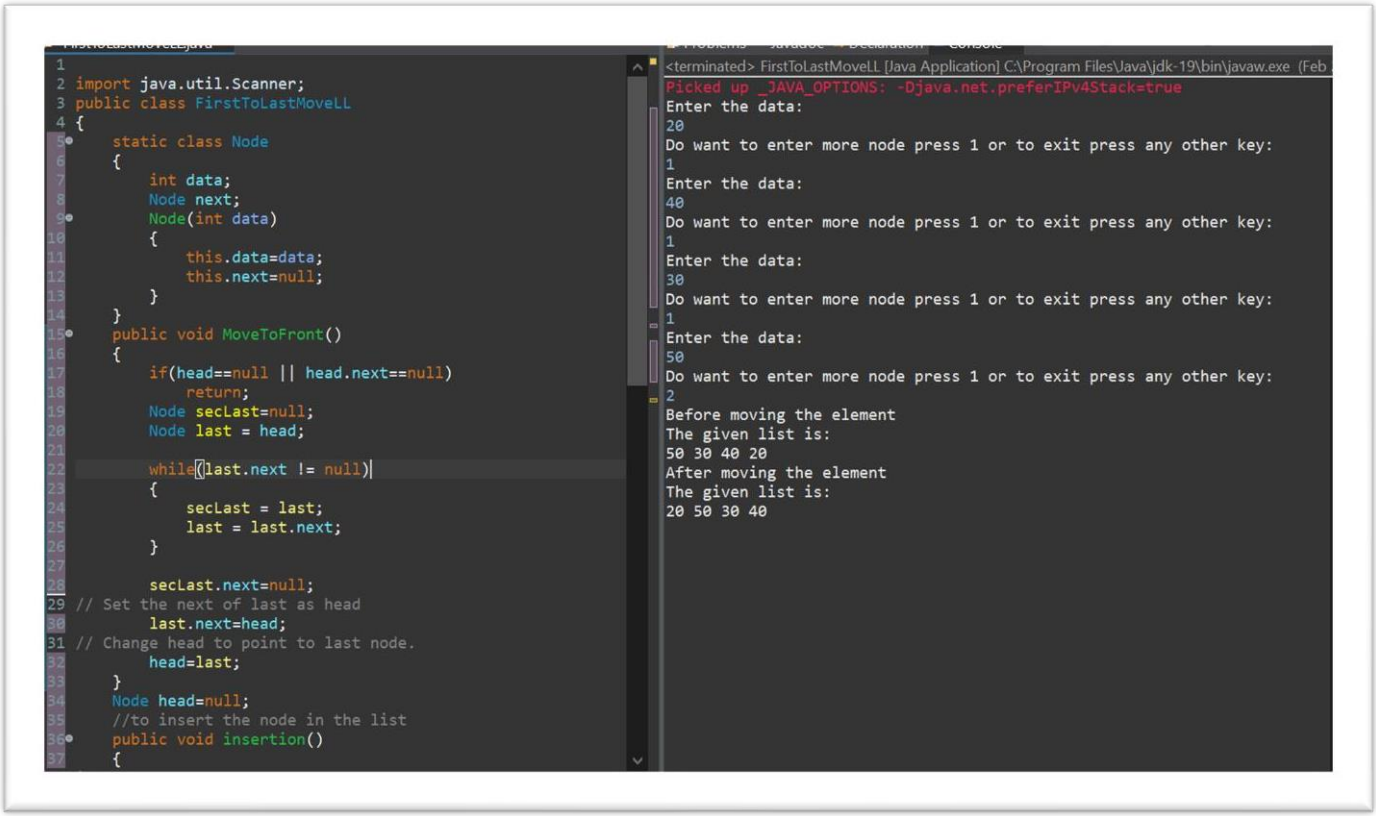
```
import  java.util.Scanner;

public class LengthOfTheSinglyLL

{

static class Node

{

int data; Node next;
```

```
Node(int data)

{

this.data=data; this.next=null;

}

}

Node head=null;

//to insert the node in the list

public void insertion()

{

int data,n;

Scanner sc=new Scanner(System.in);

do

{

System.out.println("Enter the data: "); data=sc.nextInt();

Node new_node = new Node(data);

if(head==null)

{

head=new_node;

}

else

{

new_node.next=head;

head=new_node;

}

System.out.println("Do want to enter more node press 1 or to exit press any other
key: ");
```

```
n=sc.nextInt();

}

while(n==1);

}

//to display the elemements available in the list

public void display()

{

Node temp=head; System.out.println("The given list is: "); while(temp!=null)

{

System.out.print(temp.data +" "); temp=temp.next;

}

System.out.println();

}

//To get lenght

public int getCount()

{

Node temp = head;

int count = 0;

while (temp != null)

{

count++;

temp = temp.next;

}

return count;

}
```

```
public static void main(String[] args)

{

LengthOfTheSinglyLL ll = new LengthOfTheSinglyLL(); ll.insertion();

ll.display();

System.out.println("The lenght of the list is: "+ll.getCount());

}

}
```

2. Implement a program to move last node of a singly linked list to front.



```
import java.util.Scanner;
public class FirstToLastMoveLL
{
    static class Node
    {
        int data;
        Node next;
        Node(int data)
        {
            this.data=data;
            this.next=null;
        }
    }
    public void MoveToFront()
    {
        if(head==null || head.next==null)
            return;
        Node secLast=null;
        Node last = head;

        while(last.next != null)
        {
            secLast = last;
            last = last.next;
        }

        secLast.next=null;
// Set the next of last as head
        last.next=head;
// Change head to point to last node.
        head=last;
    }
    Node head=null;
    //to insert the node in the list
    public void insertion()
    {
```

```
<terminated> FirstToLastMoveLL [Java Application] C:\Program Files\Java\jdk-19\bin\javaw.exe (Feb
Picked up _JAVA_OPTIONS: -Djava.net.preferIPv4Stack=true
Enter the data:
20
Do want to enter more node press 1 or to exit press any other key:
1
Enter the data:
40
Do want to enter more node press 1 or to exit press any other key:
1
Enter the data:
30
Do want to enter more node press 1 or to exit press any other key:
1
Enter the data:
50
Do want to enter more node press 1 or to exit press any other key:
2
Before moving the element
The given list is:
50 30 40 20
After moving the element
The given list is:
20 50 30 40
```

Code:

```
import java.util.Scanner;

public class FirstToLastMoveLL
```

```
{
static class Node
{
int data;
Node next;
Node(int data)
{
this.data=data;
this.next=null;
}
}
public void MoveToFront()
{
if(head==null || head.next==null)
return;
Node secLast=null;
Node last = head;
while(last.next != null)
{
secLast = last;
last = last.next;
}
secLast.next=null;
// Set the next of last as head
last.next=head;
```

```
// Change head to point to last node.

head=last;

}

Node head=null;

//to insert the node in the list

public void insertion()

{

int data,n;

Scanner sc=new Scanner(System.in);

do

{

System.out.println("Enter the data: ");

data=sc.nextInt();

Node new_node = new Node(data);

if(head==null)

{

head=new_node;

}

else

{

new_node.next=head;

head=new_node;

}

System.out.println("Do want to enter more node press 1 or to exit press any other
key: ");

n=sc.nextInt();
```

```
}
while(n==1);
}
public void display()
{
Node temp=head;
System.out.println("The given list is: ");
while(temp!=null)
{
System.out.print(temp.data +" ");
temp=temp.next;
}
System.out.println();
}
public static void main(String[] args)
{
FirstToLastMoveLL linkl=new FirstToLastMoveLL();
linkl.insertion();
System.out.println("Before moving the element");
linkl.display();
linkl.MoveToFront();
System.out.println("After moving the element");
linkl.display();
}
}
```