

Name : Hardik Patel
Std Id : 202103032

Exercise:

1. Write a program for the insertion of new element at the beginning of the link list.
2. Write a program for the insertion of new element at the end of the link list.
3. Write a program to perform traversal of linked list and find the maximum and minimum elements

Code:

```
package com.DSA.LAB5;
public class Exercisel {
    Node head;
    class Node {
        int data;
        Node next;
        Node(int data) {
            this.data = data;
            this.next = null;
        }
    }
    // add node at beginning of the list
    public void addFirst(int data){
        Node newNode = new Node(data);
        if(head == null){ //if list is empty assign new node
to head
            head = newNode;
            return;
        }
        newNode.next = head; //if list is not empty
        head = newNode;
    }
    // add node at last of the list
    public void addLast(int data){
        Node newNode = new Node(data);
        if(head == null){
            head = newNode;
            return;
        }
        Node lastNode = head;
        while (lastNode.next != null){
            lastNode = lastNode.next;
        }
        lastNode.next = newNode;
    }
}
```

```

public void printList() {
    if(head == null) {
        System.out.println("List is Empty");
    }
    Node currNode = head;
    while (currNode != null) {
        System.out.print(currNode.data + "->");
        currNode = currNode.next;
    }
    System.out.println("Null");
}
//find min value in the list
public void minNode() {
    Exercisel.Node current = head;
    int min;
    if(head == null) {
        System.out.println("List is empty");
    }
    else {
        min = head.data;
        while(current != null) {
            if(min > current.data) {
                min = current.data;
            }
            current = current.next;
        }
        System.out.println("Minimum value in the list: " +
min);
    }
}
//find max value in the list
public void maxNode() {
    Exercisel.Node current = head;
    int max;
    if(head == null) {
        System.out.println("List is empty");
    }
    else {
        max = head.data;
        while(current != null) {
            if(max < current.data) {
                max = current.data;
            }
            current = current.next;
        }
        System.out.println("Maximum value in the list : " +

```

```

max);
    }
}
public static void main(String[] args) {
    Exercisel list = new Exercisel();
    list.addFirst(6);    //add 6
    list.addFirst(3);    //add 3 before 6
    list.addFirst(1);    // add 1 before 3
    list.printList();
    list.addLast(8);     // add 8 after 6
    list.addLast(10);    // add 10 after 8
    list.addLast(20);    // add 20 after 10
    list.printList();
    list.minNode();
    list.maxNode();
}
}

```

Output :

```

"C:\Program Files\Java\jdk-18\bin\java.exe" "-javaa
3->6->Null
3->6->8->10->Null
Minimum value in the list: 3
Maximum value in the list : 10

Process finished with exit code 0

```

```

"C:\Program Files\Java\jdk-18\bin\java.exe" "-
1->3->6->Null
1->3->6->8->10->20->Null
Minimum value in the list: 1
Maximum value in the list : 20

Process finished with exit code 0

```